

---

# Optimal Algorithms for Stochastic Multi-Level Compositional Optimization

---

Wei Jiang<sup>1</sup> Bokun Wang<sup>2</sup> Yibo Wang<sup>1</sup> Lijun Zhang<sup>1</sup> Tianbao Yang<sup>2</sup>

## Abstract

In this paper, we investigate the problem of stochastic multi-level compositional optimization, where the objective function is a composition of multiple smooth but possibly non-convex functions. Existing methods for solving this problem either suffer from sub-optimal sample complexities or need a huge batch size. To address these limitations, we propose a Stochastic Multi-level Variance Reduction method (SMVR), which achieves the optimal sample complexity of  $\mathcal{O}(1/\epsilon^3)$  to find an  $\epsilon$ -stationary point for non-convex objectives. Furthermore, when the objective function satisfies the convexity or Polyak-Łojasiewicz (PL) condition, we propose a stage-wise variant of SMVR and improve the sample complexity to  $\mathcal{O}(1/\epsilon^2)$  for convex functions or  $\mathcal{O}(1/(\mu\epsilon))$  for non-convex functions satisfying the  $\mu$ -PL condition. The latter result implies the same complexity for  $\mu$ -strongly convex functions. To make use of adaptive learning rates, we also develop Adaptive SMVR, which achieves the same complexities but converges faster in practice. All our complexities match the lower bounds not only in terms of  $\epsilon$  but also in terms of  $\mu$  (for PL or strongly convex functions), without using a large batch size in each iteration.

## 1. Introduction

We consider the stochastic multi-level compositional optimization problem:

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) = f_K \circ f_{K-1} \circ \dots \circ f_1(\mathbf{w}), \quad (1)$$

where  $f_i : \mathbb{R}^{d_{i-1}} \mapsto \mathbb{R}^{d_i}$ ,  $i = 1, \dots, K$  (with  $d_K = 1$  and  $d_0 = d$ ). Only noisy evaluations of each layer function

<sup>1</sup>National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China <sup>2</sup>Department of Computer Science, The University of Iowa, Iowa City, USA. Correspondence to: Lijun Zhang <zhanglj@lamda.nju.edu.cn>, Tianbao Yang <tianbao.yang@uiowa.edu>.

$f_i(\cdot; \xi)$  and its gradient  $\nabla f_i(\cdot; \xi)$  can be accessed, where  $\xi$  denotes a sample drawn from the oracle such that:

$$\mathbb{E}_\xi [f_i(\cdot; \xi)] = f_i(\cdot), \quad \mathbb{E}_\xi [\nabla f_i(\cdot; \xi)] = \nabla f_i(\cdot).$$

In machine learning,  $\mathbf{w}$  often represents the parameter of a predictive model, and  $F$  denotes the loss of that model, with  $\xi$  representing a training sample (or a batch of samples). Problem (1) finds its application in many tasks, such as reinforcement learning (Dann et al., 2014), robust learning (Li et al., 2021b), multi-step model-agnostic meta-learning (Ji et al., 2020), risk-averse portfolio optimization (Bruno et al., 2016; Shapiro et al., 2021) and risk management (Cole et al., 2017; Dentcheva et al., 2017).

Our goal is to solve problem (1) with the optimal sample complexity, which is a commonly-used measure in stochastic optimization. Sample complexity characterizes the number of samples needed to find an  $\epsilon$ -stationary point for non-convex functions, i.e.,  $\|\nabla F(\mathbf{w})\| \leq \epsilon^1$ , or an  $\epsilon$ -optimal point for convex functions, i.e.,  $F(\mathbf{w}) - \inf_{\mathbf{w}} F(\mathbf{w}) \leq \epsilon$ . Problem (1) reduces to the classic one-level stochastic optimization problem when  $K = 1$ , and it is also known as the two-level compositional optimization for  $K = 2$ .

For one-level and two-level non-convex problems, there exist single-loop algorithms, such as STORM (Cutkosky & Orabona, 2019) and RECOVER (Qi et al., 2021), that can achieve the optimal  $\mathcal{O}(1/\epsilon^3)$  sample complexity for finding an  $\epsilon$ -stationary solution without using large batches. However, for multi-level problems, the errors of Jacobian and function value estimators accumulate with the level becoming deeper, making the problem much harder. Existing multi-level methods either suffer from sub-optimal complexities (Yang et al., 2019; Balasubramanian et al., 2021; Chen et al., 2021) or require a huge and increasing batch size (Zhang & Xiao, 2021). When the objective function is convex or strongly convex, Zhang & Lan (2021) prove a sample complexity of  $\mathcal{O}(1/\epsilon^2)$  or  $\mathcal{O}(1/(\mu^2\epsilon))$ . However, their analysis requires that each layer function  $f_i$  is monotone and convex, and their complexity for  $\mu$ -strongly convex

<sup>1</sup>In some literature, the measure  $\|\nabla F(\mathbf{w})\|^2 \leq \epsilon$  is used instead. Note that the complexity  $\mathcal{O}(1/\epsilon^\alpha)$  for  $\|\nabla F(\mathbf{w})\|^2 \leq \epsilon$  implies a complexity  $\mathcal{O}(1/\epsilon^{2\alpha})$  for  $\|\nabla F(\mathbf{w})\| \leq \epsilon$  (Zhang & Xiao, 2021). For fair comparison, we would align the complexity measure of each method under the same criteria  $\|\nabla F(\mathbf{w})\| \leq \epsilon$ , when discussing related algorithms.

Table 1. Summary of results for finding an  $\epsilon$ -stationary or  $\epsilon$ -optimal point. Here, CVX means convex, Mono. & CVX means each layer function is monotone and convex, SC means  $\mu$ -strongly convex and PL means the  $\mu$ -PL condition (weaker than  $\mu$ -strongly convex).

| METHOD                               | ASSUMPTIONS          | COMPLEXITY                                     | BATCH SIZE                |
|--------------------------------------|----------------------|--|---------------------------|
| A-TSCGD (YANG ET AL., 2019)          | SMOOTH               | $\mathcal{O}\left(1/\epsilon^{(7+K)/2}\right)$ | $\mathcal{O}(1)$          |
| A-TSCGD (YANG ET AL., 2019)          | SMOOTH + SC          | $\mathcal{O}\left(1/\epsilon^{(3+K)/4}\right)$ | $\mathcal{O}(1)$          |
| NLASG (BALASUBRAMANIAN ET AL., 2021) | SMOOTH               | $\mathcal{O}(1/\epsilon^4)$                    | $\mathcal{O}(1)$          |
| SCSC (CHEN ET AL., 2021)             | SMOOTH               | $\mathcal{O}(1/\epsilon^4)$                    | $\mathcal{O}(1)$          |
| NESTED-SPIDER (ZHANG & XIAO, 2021)   | SMOOTH               | $\mathcal{O}(1/\epsilon^3)$                    | $\mathcal{O}(1/\epsilon)$ |
| SSD (ZHANG & LAN, 2021)              | SMOOTH + MONO. & CVX | $\mathcal{O}(1/\epsilon^2)$                    | $\mathcal{O}(1)$          |
| SSD (ZHANG & LAN, 2021)              | SMOOTH + SC          | $\mathcal{O}(1/(\mu^2\epsilon))$               | $\mathcal{O}(1)$          |
| <b>SMVR (THIS WORK)</b>              | SMOOTH               | $\mathcal{O}(1/\epsilon^3)$                    | $\mathcal{O}(1)$          |
| <b>STAGE-WISE SMVR (THIS WORK)</b>   | SMOOTH + CVX         | $\mathcal{O}(1/\epsilon^2)$                    | $\mathcal{O}(1)$          |
| <b>STAGE-WISE SMVR (THIS WORK)</b>   | SMOOTH + PL          | $\mathcal{O}(1/(\mu\epsilon))$                 | $\mathcal{O}(1)$          |

function is non-optimal in terms of  $\mu$  (Agarwal et al., 2012).

Hence, a fundamental question to be addressed is:

*Can we solve stochastic multi-level compositional problems with optimal complexities for non-convex, convex and strongly convex functions without using a large batch size?*

We give an affirmative answer to this question by proposing an optimal algorithm named Stochastic Multi-level Variance Reduction (SMVR). By using the variance reduction technique to estimate the Jacobians and function values in each level, our algorithm achieves the optimal sample complexity of  $\mathcal{O}(1/\epsilon^3)$  for non-convex functions (Arjevani et al., 2019). Central to the algorithmic design and analysis are: (i) the variance reduction is applied to both Jacobians and function values, which is different from most existing works (Yang et al., 2019; Balasubramanian et al., 2021; Chen et al., 2021); (ii) the Jacobian estimators are updated with a projection to ensure that errors of gradient estimators can be bounded regardless of the depth of the problem. When the objective function is convex or satisfies the  $\mu$ -PL condition (weaker than strong convexity), we develop a stage-wise version of our method and improve the complexity to  $\mathcal{O}(1/\epsilon^2)$  or  $\mathcal{O}(1/(\mu\epsilon))$ , which matches the corresponding lower bounds (Agarwal et al., 2012). The key in the analysis is to prove that the errors of Jacobian and function value estimators decrease in a stage-wise manner. Finally, to take advantage of adaptive learning rates, we also design an adaptive version of the SMVR method and prove the same rates. Adaptive SMVR performs better in practice and avoids tuning the learning rate manually.

Compared with existing multi-level methods, this paper enjoys the following advantages:

1. We obtain the optimal sample complexity of  $\mathcal{O}(1/\epsilon^3)$

for non-convex objectives, which is better than existing multi-level methods (Yang et al., 2019; Balasubramanian et al., 2021; Chen et al., 2021). Although Zhang & Xiao (2021) achieve the same rate, their method uses a large and increasing batch size at the order of  $\mathcal{O}(1/\epsilon)$ , which is impractical to use.

2. We achieve the optimal complexity of  $\mathcal{O}(1/\epsilon^2)$  and  $\mathcal{O}(1/(\mu\epsilon))$  for convex and strongly convex functions respectively. Compared with Zhang & Lan (2021), we do not require each layer function  $f_i$  is monotone and convex, and have a better dependence on  $\mu$  for  $\mu$ -strongly convex functions.
3. We develop Adaptive SMVR method to make use of adaptive learning rates, which enjoys the same complexity but converges faster in practice.

A comparison between our results and existing multi-level methods is shown in Table 1 and empirical results on practical problems demonstrate the effectiveness of our method.

## 2. Related Work

This section briefly reviews related work on stochastic two-level and multi-level compositional optimization problems.

### 2.1. Two-Level Compositional Optimization

Wang et al. (2017a) first introduce stochastic compositional gradient descent (SCGD) to minimize a composition of two-level expected-value functions. The SCGD method adopts two step size sequences in different time scales to update the decision variable and inner function separately. When the inner function is smooth, their method achieves a complexity of  $\mathcal{O}(1/\epsilon^7)$  for non-convex objectives,  $\mathcal{O}(1/\epsilon^{3.5})$  for convex functions, and  $\mathcal{O}(1/(\mu^{14/4}\epsilon^{5/4}))$  for  $\mu$ -strongly convex functions. In a subsequent work (Wang et al., 2017b), the accelerated stochastic compositional proximal gradi-

ent (ASC-PG) is proposed to improve the complexity to  $\mathcal{O}(1/\epsilon^{4.5})$ ,  $\mathcal{O}(1/\epsilon^2)$ , and  $\mathcal{O}(1/\epsilon)$  for non-convex, convex and strongly convex functions, respectively.

Instead of using two-timescale step sizes, a single-timescale method called Nested Averaged Stochastic Approximation (NASA) has been developed by Ghadimi et al. (2020) which achieves the complexity of  $\mathcal{O}(1/\epsilon^4)$  for non-convex objectives. With the increasing popularity of variance reduction techniques for one-level stochastic optimization such as SARAH (Nguyen et al., 2017), SPIDER (Fang et al., 2018), SpiderBoost (Wang et al., 2018) and STORM (Cutkosky & Orabona, 2019), variance reduced algorithms are also developed for two-level compositional problems with improved rates under a slightly stronger smoothness assumption (Huo et al., 2018; Yuan et al., 2019; Zhang & Xiao, 2019a;b; Liu et al., 2021). The optimal  $\mathcal{O}(1/\epsilon^3)$  sample complexity is achieved by Yuan et al. (2019) based on SARAH and by Zhang & Xiao (2019b) based on SPIDER with a large batch size. Later, Yuan & Hu (2020) develop an algorithm named STORM-Compositional, which leads to an  $\mathcal{O}(1/\epsilon^3)$  complexity using mini-batches. To avoid using batches, Qi et al. (2021) propose a method based on STORM and obtain the same rate. However, these two-level methods can not be extended to multi-level optimization problems directly.

## 2.2. Multi-Level Compositional Optimization

Yang et al. (2019) first investigate the problem of multi-level optimization and develop accelerated  $T$ -level stochastic compositional gradient descent (A-TSCGD). By using an extrapolation-interpolation scheme, their method achieves a sample complexity of  $\mathcal{O}(1/\epsilon^{(7+K)/2})$  for  $K$ -level problems. This rate is improved to  $\mathcal{O}(1/\epsilon^{(3+K)/4})$  for strongly convex functions. Later, Balasubramanian et al. (2021) propose a nested linearized averaging stochastic gradient method (NLASG), which extends the NASA (Ghadimi et al., 2020) algorithm to the general  $K \geq 1$  setting and achieves a sample complexity of  $\mathcal{O}(1/\epsilon^4)$ . In a concurrent work, Chen et al. (2021) come up with a stochastically corrected stochastic compositional gradient method (SCSC), which uses a technique similar to STORM to estimate the function values at each level, and also has a sample complexity of  $\mathcal{O}(1/\epsilon^4)$ .

Recently, Zhang & Xiao (2021) present the Nested-SPIDER method, which uses nested variance reduction to approximate the gradient and improves the sample complexity to  $\mathcal{O}(1/\epsilon^3)$ . However, their method requires a large and increasing batch size at the order of  $\mathcal{O}(1/\epsilon)$ . In the first iteration of each stage, the batch size even has to be set as large as  $\mathcal{O}(1/\epsilon^2)$ . No complexities are provided for convex and strongly convex functions. Later, Zhang & Lan (2021) prove that the sample complexity can be improved to  $\mathcal{O}(1/\epsilon^2)$  when every layer function  $f_i$  is monotone and

convex, using a general stochastic sequential dual method (SSD). The complexity is further reduced to  $\mathcal{O}(1/(\mu^2\epsilon))$  for  $\mu$ -strongly convex functions. However, their method requires strong assumptions, i.e., layer-wise convexity and monotonicity. In contrast, our method only requires the overall objective function to be convex or strongly convex to achieve the same complexity for convex functions and an even better complexity for strongly convex functions.

## 3. The Proposed Method

We first discuss the main challenge in solving multi-level compositional optimization problems. Then, we develop an optimal method for non-convex objectives. Finally, we explore additional conditions to further improve the sample complexity.

### 3.1. Notations and Assumptions

Let  $\xi$  represent some random variable (in practice, it represents a training sample or a batch of samples drawn from the oracle) and  $\|\cdot\|$  denote the Euclidean norm of a vector. For simplicity, we use  $\Pi_{L_f}$  to denote the projection onto the ball with radius  $L_f$ , i.e.,

$$\Pi_{L_f}(\mathbf{x}) = \underset{\|\mathbf{w}\| \leq L_f}{\operatorname{argmin}} \|\mathbf{w} - \mathbf{x}\|^2.$$

We give the definition of sample complexity below.

**Definition 1.** *The sample complexity is the number of samples needed to find a point satisfying  $\mathbb{E}[\|\nabla F(\mathbf{w})\|] \leq \epsilon$  ( $\epsilon$ -stationary) or  $\mathbb{E}[F(\mathbf{w}) - \inf_{\mathbf{w}} F(\mathbf{w})] \leq \epsilon$  ( $\epsilon$ -optimal).*

Moreover, we make the following assumptions throughout the paper, which are commonly used in the studies of stochastic compositional optimization (Wang et al., 2017b;a; Yuan et al., 2019; Zhang & Xiao, 2019b; 2021). We treat the parameters  $L_f$ ,  $L_J$ ,  $\sigma_f$ ,  $\sigma_J$ ,  $\mathcal{L}_f$ ,  $\mathcal{L}_J$  below as global constants.

**Assumption 1.** (*Smoothness and Lipschitz continuity*) *All functions  $f_1, \dots, f_K$  are  $L_f$ -Lipschitz continuous and their Jacobians  $\nabla f_1, \dots, \nabla f_K$  are  $L_J$ -Lipschitz continuous. Note that this implies the objective function  $F(\mathbf{w})$  is  $L_F$  smooth, where  $L_F = L_f^{2K-1} L_J \sum_{i=1}^K \frac{1}{L_f^i}$ .*

**Assumption 2.** (*Bounded variance*) *For  $1 \leq i \leq K$ :*

$$\begin{aligned} \mathbb{E}_{\xi_t^i} [f_i(\mathbf{x}; \xi_t^i)] &= f_i(\mathbf{x}), \\ \mathbb{E}_{\xi_t^i} [\nabla f_i(\mathbf{x}; \xi_t^i)] &= \nabla f_i(\mathbf{x}), \\ \mathbb{E}_{\xi_t^i} [\|f_i(\mathbf{x}; \xi_t^i) - f_i(\mathbf{x})\|^2] &\leq \sigma_f^2, \\ \mathbb{E}_{\xi_t^i} [\|\nabla f_i(\mathbf{x}; \xi_t^i) - \nabla f_i(\mathbf{x})\|^2] &\leq \sigma_J^2, \end{aligned}$$

where  $\{\xi_t^i\}_{i=1}^K$  are mutually independent.

**Assumption 3.** (Mean-squared smoothness)

$$\begin{aligned} \mathbb{E}_{\xi_t^i} \left[ \|f_i(\mathbf{x}; \xi_t^i) - f_i(\mathbf{y}; \xi_t^i)\|^2 \right] &\leq \mathcal{L}_f^2 \|\mathbf{x} - \mathbf{y}\|^2, \\ \mathbb{E}_{\xi_t^i} \left[ \|\nabla f_i(\mathbf{x}; \xi_t^i) - \nabla f_i(\mathbf{y}; \xi_t^i)\|^2 \right] &\leq \mathcal{L}_J^2 \|\mathbf{x} - \mathbf{y}\|^2. \end{aligned}$$

**Assumption 4.**  $F_* = \inf_{\mathbf{w}} F(\mathbf{w}) \geq -\infty$  and  $F(\mathbf{w}_1) - F_* \leq \Delta_F$  for the initial solution  $\mathbf{w}_1$ .

### 3.2. The Challenge

Compared with one-level problems, the main dilemma in multi-level optimization is that we can not obtain the unbiased gradient of the function  $F$ . Take a two-level compositional problem as an example. The objective function can be written as:  $F(\mathbf{w}) = f \circ g(\mathbf{w})$  and its gradient is:

$$\nabla F(\mathbf{w}) = \nabla g(\mathbf{w}) \cdot \nabla f(g(\mathbf{w})).$$

Although we can access to the unbiased estimation of each layer function and its gradient, i.e.,  $\mathbb{E}_{\xi_1} [g(\mathbf{x}; \xi_1)] = g(\mathbf{x})$ ,  $\mathbb{E}_{\xi_2} [f(\mathbf{x}; \xi_2)] = f(\mathbf{x})$  and  $\mathbb{E}_{\xi_2} [\nabla f(\mathbf{x}; \xi_2)] = \nabla f(\mathbf{x})$ , it is still challenging to obtain an unbiased estimation of the gradient  $\nabla f(g(\mathbf{w}))$ . This is because the expectation over  $\xi_1$  cannot be moved inside of  $\nabla f$  such that:

$$\mathbb{E}_{\xi_1, \xi_2} [\nabla f(g(\mathbf{w}; \xi_1); \xi_2)] \neq \nabla f(g(\mathbf{w})).$$

Due to the same reason, it is also difficult to get the unbiased estimation of the function value:

$$\mathbb{E}_{\xi_1, \xi_2} [f(g(\mathbf{w}; \xi_1); \xi_2)] \neq f(g(\mathbf{w})).$$

The above challenge motivates us to use the variance reduced estimator to have a better evaluation of both function values and Jacobians of each level to ensure that the estimation errors reduce over time.

However, variance reduced estimators used in two-level optimization problems (Qi et al., 2021) can not be applied to multi-level directly, because the error might blow up as the depth increases if the estimators of Jacobians are not bounded. To handle this issue, Zhang & Xiao (2021) use an extremely small step size and re-estimate the function values and Jacobians of all levels after several iterations with a large batch size. However, this method inevitably introduces large batches (as large as  $\mathcal{O}(1/\epsilon^2)$ ) at the beginning of each stage, and since they use SPIDER (Fang et al., 2018) as their estimator, their method requires a batch size of  $\mathcal{O}(1/\epsilon)$  at other iterations. To avoid using large batches, our method uses STORM (Cutkosky & Orabona, 2019) estimator and projects gradients onto a ball to ensure the Jacobians can be well bounded so that the error of the gradient estimator does not blow up.

### 3.3. Stochastic Multi-Level Variance Reduction

Now, we present the proposed algorithm – Stochastic Multi-level Variance Reduction method (SMVR) for solving prob-

---

#### Algorithm 1 SMVR

---

**Input:** time step  $T$ , initial points  $(\mathbf{w}_1, \mathbf{u}_1, \mathbf{v}_1)$ , parameter  $c$  and learning rate sequence  $\{\eta_t\}$   
**for** time step  $t = 1$  **to**  $T$  **do**  
 Set  $\mathbf{u}_t^0 = \mathbf{w}_t, \beta_t = c\eta_{t-1}^2$   
**for** level  $i = 1$  **to**  $K$  **do**  
 Sample  $\xi_t^i$   
 Compute estimator  $\mathbf{u}_t^i$  according to (2)  
 Compute estimator  $\mathbf{v}_t^i$  according to (3)  
**end for**  
 Update gradient estimation:  $\mathbf{v}_t = \prod_{i=1}^K \mathbf{v}_t^i$   
 Update the weight:  $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \mathbf{v}_t$   
**end for**  
 Choose  $\tau$  uniformly at random from  $\{1, \dots, T\}$   
 Return  $(\mathbf{w}_\tau, \mathbf{u}_\tau, \mathbf{v}_\tau)$

---

lem (1). The goal of our algorithm is to find an  $\epsilon$ -stationary point with low sample complexity. As mentioned before, the main difficulty is that we can not obtain an unbiased estimation of the gradients and inner function values in the multi-level setting. We note that, in the one-level problem, STORM uses a momentum-based variance reduction method to evaluate the gradient:

$$\begin{aligned} \mathbf{d}_t &= (1 - \beta_t) \mathbf{d}_{t-1} + \beta_t \nabla f(\mathbf{x}_t; \xi_t) \\ &\quad + (1 - \beta_t) (\nabla f(\mathbf{x}_t; \xi_t) - \nabla f(\mathbf{x}_{t-1}; \xi_t)). \end{aligned}$$

This technique reduces the variance of the estimated value and obtains the optimal rate. Inspired by STORM, we apply similar variance reduced estimators in each level to approximate the gradient more accurately.

The proposed method is described in Algorithm 1. In each time step  $t$ , we use two sequence  $\mathbf{u}_t^i$  and  $\mathbf{v}_t^i$  to estimate the function value and the gradient in level  $i$ . To estimate the function value, we use a nested STORM estimator, i.e.,

$$\begin{aligned} \mathbf{u}_t^i &= (1 - \beta_t) \mathbf{u}_{t-1}^i + \beta_t f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) \\ &\quad + (1 - \beta_t) (f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i)). \end{aligned} \quad (2)$$

This can be interpreted as that  $\mathbf{u}_t^i$  is a STORM estimator of  $f_i(\mathbf{u}_t^{i-1})$ . For estimating the Jacobians, we use the nested STORM estimator followed by a projection, i.e.,

$$\begin{aligned} \mathbf{v}_t^i &= \Pi_{L_f} \left[ (1 - \beta_t) \mathbf{v}_{t-1}^i + \beta_t \nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) \right. \\ &\quad \left. + (1 - \beta_t) (\nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i)) \right]. \end{aligned} \quad (3)$$

The projection operation is to ensure the error of the stochastic gradient estimator can be bounded; otherwise, they may blow up as the level becomes deeper. That is to say, on one hand, we want to enjoy variance reduction of the estimator (since true gradients are in the projected domain, projection does not hinder the analysis); on the other hand we do not want the variance of estimator accumulates over multiple

**Algorithm 2** Stage-wise SMVR

---

**Input:** initial points  $(\mathbf{w}_0, \mathbf{u}_0, \mathbf{v}_0)$ , parameter  $c$   
**for** stage  $s = 1$  **to**  $S$  **do**  
     Set  $\eta_s$  and  $T_s$  according to Lemma 1  
      $\mathbf{w}_s, \mathbf{u}_s, \mathbf{v}_s = \text{SMVR}$  (with  $T_s, (\mathbf{w}_{s-1}, \mathbf{u}_{s-1}, \mathbf{v}_{s-1}), c$   
         and  $\eta_s$ )  
**end for**  
 Return  $\mathbf{w}_S$

---

levels ( $\mathbf{v}_i$  is bounded after projection). Hence, projection on the Jacobian estimator is a perfect solution. After the gradient of each level is evaluated, we use the chain rule to calculate the estimated gradient of the objective function, i.e.,  $\mathbf{v}_t = \mathbf{v}_t^1 \mathbf{v}_t^2 \cdots \mathbf{v}_t^K$  and apply gradient descent to update the decision variable  $\mathbf{w}_t$  at the end of each time step.

Note that in the first iteration, we evaluate the function value and gradient of each level simply as  $\mathbf{u}_1^i = f(\mathbf{u}_1^{i-1}; \xi_1^i)$  and  $\mathbf{v}_1^i = \nabla f_i(\mathbf{u}_1^{i-1}; \xi_1^i)$ . Our algorithm does not need to use batches in any iterations. Of course, it supports mini-batches, and  $\xi_t^i$  in the algorithm can represent a training sample or a batch of samples. Next, we show the sample complexity of the proposed method. Due to space limitations, all the proofs are deferred to the appendix. We define the constant  $L_1 = \max \left\{ 1, KL_f^{2(K-1)}, KL_F^2, 2K(\sigma_J^2 + \sigma_f^2), 2(L_J^2 + L_f^2) \left( 2K + 2K\sigma_f^2 \right) \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \right\}$ .

**Theorem 1.** *If we set  $c = 10L_1^2$ ,  $\eta_t = (a+t)^{-1/3}$  and  $a = (20L_1^3)^{3/2}$ , our algorithm finds an  $\epsilon$  stationary point in  $\mathcal{O}(1/\epsilon^3)$  iterations.*

**Remark:** The complexity is at the order of  $\mathcal{O}(1/\epsilon^3)$ , which matches the lower bound in one-level setting (Arjevani et al., 2019), implying multi-level setting does not make the problem much harder. Our SMVR method uses decreasing learning rates and avoids using batches in any iteration, which is more practical to implement compared with the existing method which requires huge batch size and changing the batch size over time (Zhang & Xiao, 2021).

### 3.4. Faster Convergence under Stronger Conditions

In this section, we explore whether other assumptions could be utilized to further improve the sample complexity. We develop a variant algorithm named Stage-wise SMVR, which achieves better complexity when the objective function satisfies the PL condition or convexity.

The new algorithm is a multi-stage version of the SMVR method, summarized in Algorithm 2. Instead of decreasing the learning rate  $\eta_t$  polynomially, we decrease  $\eta$  and  $\beta$  after each stage and increase the number of iterations per stage. At the end of each stage, the algorithm save the

output  $\mathbf{w}_s, \mathbf{u}_s, \mathbf{v}_s$ , which are used for restarting in the next stage. With these modifications, we can obtain a better convergence guarantee under the PL condition or convexity.

First, we investigate the case that the objective function satisfies the PL condition, which is a commonly used condition in the literature (Charles & Papailiopoulos, 2018; Nouiehed et al., 2019; Xie et al., 2020; Chewi et al., 2020). We first introduce the definition of the PL condition.

**Definition 2.**  $F(\mathbf{w})$  satisfies the  $\mu$ -PL condition if there exists  $\mu > 0$  such that:

$$2\mu(F(\mathbf{w}) - F_*) \leq \|\nabla F(\mathbf{w})\|^2.$$

Note that a function can be non-convex and still satisfy the PL condition. Also, the PL condition is weaker than strong convexity (Karimi et al., 2016). With this condition, we can prove that the error of function estimator  $\mathbf{u}_s$  and gradient estimator  $\mathbf{v}_s$  would decrease after each stage.

**Lemma 1.** *Define  $\epsilon_1 = \frac{8L_1}{\mu}$  and  $\epsilon_s = \frac{\epsilon_1}{2^{s-1}}$ , with  $\beta_1 = \frac{1}{2L_1}$ ,  $T_1 = \max \left\{ 4L_1K(\sigma_f^2 + \sigma_J^2), 2\sqrt{2L_1}\Delta_F \right\}$ ,  $\beta_s = \frac{\mu\epsilon_{s-1}}{L_2}, T_s = \max \left\{ \frac{4L_2^{3/2}}{\mu\epsilon_{s-1}}, \frac{4L_2}{\mu^{3/2}\sqrt{\epsilon_{s-1}}} \right\}$ ,  $c = 16L_1^2$ ,  $\eta_s = \sqrt{\beta_s/c}$  and  $L_2 = 64L_1^2$ , the output of Algorithm 2 at each stage satisfies:*

$$E[F(\mathbf{w}_s) - F_*] \leq \epsilon_s;$$

$$\sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_s^{i-1}) - \mathbf{u}_s^i\|^2 + \|\mathbf{v}_s^i - \nabla f_i(\mathbf{u}_s^{i-1})\|^2 \right] \leq \mu\epsilon_s.$$

The above lemma shows that the objective gap  $F(\mathbf{w}_s) - F_*$  is halved after each stage. So, after  $S = \log_2 \left( \frac{2\epsilon_1}{\epsilon} \right)$  stages, the output of our method satisfies  $F(\mathbf{w}_S) - F_* \leq \epsilon$ . Based on Lemma 1, we prove the convergence rate of our algorithm in the following theorem.

**Theorem 2.** *Assume  $F(\mathbf{w})$  satisfies the  $\mu$ -PL condition. Stage-wise SMVR attains an  $\epsilon$ -optimal point with a sample complexity of  $\mathcal{O}(1/(\mu\epsilon))$ .*

If the objective function satisfies the convexity rather than PL condition, our method can still utilize this property to improve the sample complexity, as stated below.

**Theorem 3.** *Assume  $F(\mathbf{w})$  is convex and  $\|x^*\| \leq D$ , where  $x^*$  denote an optimal solution. Our algorithm attains an  $\epsilon$ -optimal point with a complexity of  $\mathcal{O}(1/\epsilon^2)$ .*

**Remark:** Our Stage-wise SMVR method behaves optimally when the objective function enjoys the PL condition or convexity. For smooth and convex functions, our method matches the  $\mathcal{O}(1/\epsilon^2)$  lower bound for this problem (Agarwal et al., 2012). When it comes to the PL condition, we note that there exists  $\mathcal{O}(1/(\mu\epsilon))$  lower bound for the  $\mu$ -strongly convex setting (Agarwal et al., 2012), which is a

special case of the PL condition, thus proving our method is optimal. Compared with existing results (Zhang & Lan, 2021), our analysis requires weaker assumptions and enjoys a better and optimal dependence in terms of  $\mu$ .

#### 4. SMVR with Adaptive Learning Rates

In this section, we show that the proposed method can be extended to adaptive learning rates and obtains the same sample complexity. Adaptive learning rates are widely used in stochastic optimization problems, and many successful methods have been proposed, such as AdaGrad (Duchi et al., 2010), Adam (Kingma & Ba, 2015), AMSGrad (Reddi et al., 2018), AdaBound (Luo et al., 2019), etc. However, it remains less investigated in the stochastic multi-level literature. Inspired by the above methods, we develop an adaptive version of our method, named Adaptive SMVR. To use adaptive learning rates, we revise the weight update step in Algorithm 1 as follows:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \frac{\eta_t}{\sqrt{\mathbf{h}_t + \delta}} \mathbf{v}_t, \quad (4)$$

where  $\delta > 0$  is a parameter to avoid dividing zero and the parameter  $\mathbf{h}_t$  can take following forms:

$$\begin{aligned} \text{AdaGrad-type: } \quad & \mathbf{h}_t = \frac{1}{t} \sum_{i=1}^t \mathbf{v}_i^2 \\ \text{Adam-type: } \quad & \mathbf{h}_t = (1 - \beta'_t) \mathbf{h}_{t-1} + \beta'_t \mathbf{v}_t^2 \\ \text{AMSGrad-type: } \quad & \mathbf{h}'_t = (1 - \beta'_t) \mathbf{h}'_{t-1} + \beta'_t \mathbf{v}_t^2, \\ & \mathbf{h}_t = \max(\mathbf{h}_{t-1}, \mathbf{h}'_t) \\ \text{AdaBound-type: } \quad & \mathbf{h}'_t = (1 - \beta'_t) \mathbf{h}'_{t-1} + \beta'_t \mathbf{v}_t^2, \\ & \mathbf{h}_t = \Pi_{[1/c_u^2, 1/c_l^2]}[\mathbf{h}'_t] \end{aligned} \quad (5)$$

where  $c_l \leq c_u$  and  $\Pi_{[a,b]}$  projects the input into the range  $[a, b]$ . Inspired by the recent study of Adam-style methods (Guo et al., 2021), we can give the sample complexity of the Adaptive SMVR in Theorem 4 using similar analysis.

**Theorem 4.** *If we choose  $c = 10L_3^2$ ,  $\eta_t = (a + t)^{-1/3}$  and  $a = (20L_3^3)^{3/2}$ , Adaptive SMVR with learning rate defined in (4) and (5), can obtain a stationary point in  $\mathcal{O}(1/\epsilon^3)$  iterations, where  $L_3$  is a constant indicated in the proof.*

**Remark:** The sample complexity is still at the order of  $\mathcal{O}(1/\epsilon^3)$ , when the adaptive learning rate is used. Our adaptive SMVR changes the learning rate automatically, which reduces the need to tune hyper-parameters manually. When the objective function satisfies the convexity or PL condition, Theorem 2 and Theorem 3 can be easily extended to the adaptive version with the same sample complexity.

## 5. Experiments

In this section, we conduct numerical experiments to evaluate the performance of the proposed method over three different tasks. We compare our method with existing multi-level algorithms, including A-TSCGD (Yang et al., 2019), NLAGS (Balasubramanian et al., 2021), Nested-SPIDER (Zhang & Xiao, 2021) and SCSC (Chen et al., 2021). For our method, the parameter  $\beta$  is searched from the set  $\{0.1, 0.5, 0.9\}$ . For other methods, we choose the hyper-parameters suggested in the original papers or use grid search to select the best hyper-parameters. When it comes to the learning rate, we tune it from the range  $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1\}$ . As for the projection operation  $\Pi_{L_f}$ , we can simply set  $L_f$  as a large value and we provide a sensitivity analysis in terms of tuning  $L_f$  in the first experiment.

### 5.1. Risk-Averse Portfolio Optimization

We first consider the risk-averse portfolio optimization problem. Suppose we have  $d$  assets to invest during each time step  $\{1, \dots, T\}$ , and  $r_t \in \mathbb{R}^d$  denotes the payoff of  $d$  assets in the time step  $t$ . Our goal is to maximize the return of the investment and minimize the risk at the same time. One useful formulation is the mean-deviation risk-averse optimization problem (Shapiro et al., 2021), where the risk is defined as the standard deviation. This mean-deviation model is widely used in practice and often used to conduct experiments in multi-level optimization (Yang et al., 2019; Zhang & Lan, 2021). The problem can be formulated as:

$$\max_{x \in \mathcal{X}} \frac{1}{T} \sum_{t=1}^T \langle r_t, x \rangle - \lambda \sqrt{\frac{1}{T} \sum_{t=1}^T (\langle r_t, x \rangle - \bar{r}, x)^2},$$

where  $\bar{r} = \sum_{t=1}^T r_t$  and decision variable  $x$  denotes the investment quantity vector in  $d$  assets. This problem is a three-level stochastic compositional optimization problem, and each layer can be represented as:

$$\begin{aligned} f_1(x) &= \left( \frac{1}{T} \sum_{t=1}^T \langle r_t, x \rangle, x \right), \\ f_2(y, x) &= \left( y, \frac{1}{T} \sum_{t=1}^T (\langle r_t, x \rangle - y)^2 \right), \\ f_3(z_1, z_2) &= -z_1 + \lambda \sqrt{z_2}. \end{aligned}$$

In the experiment, we test different methods on real-world datasets Industry-10, Industry-12, Industry-17 and Industry-30 from Keneth R. French Data Library<sup>2</sup>. These datasets contain 10, 12, 17 and 30 industrial assets payoff over 25105 consecutive periods, respectively. Following Zhang & Xiao (2021), we set parameter  $\lambda = 0.2$ .

<sup>2</sup><https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/>

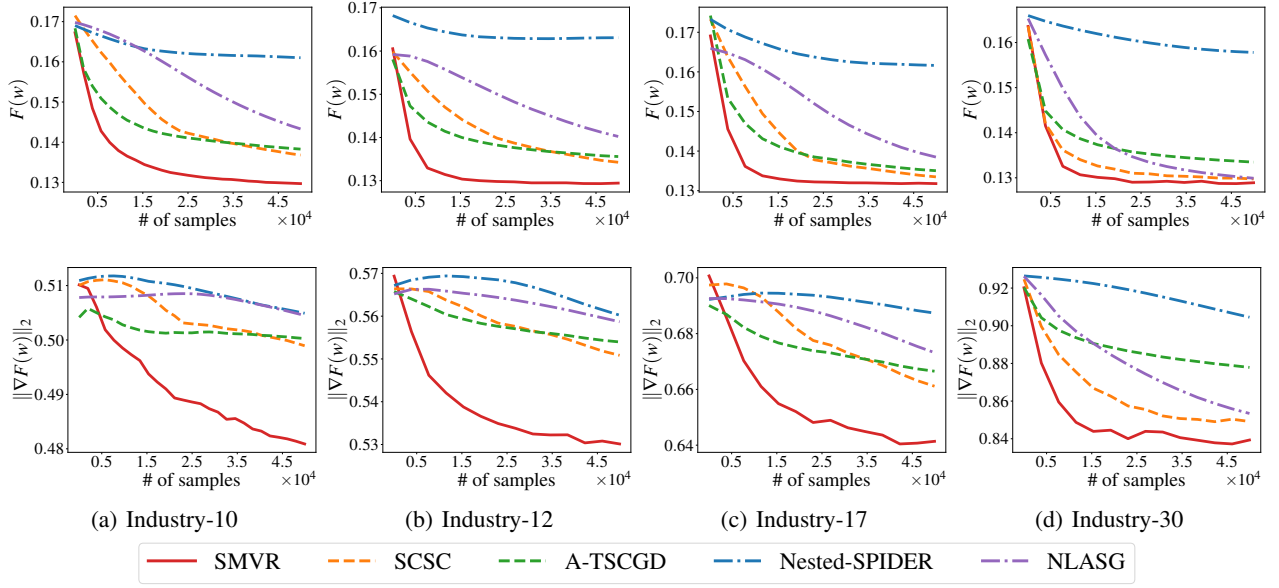


Figure 1. Results for Risk-Averse Portfolio Optimization.

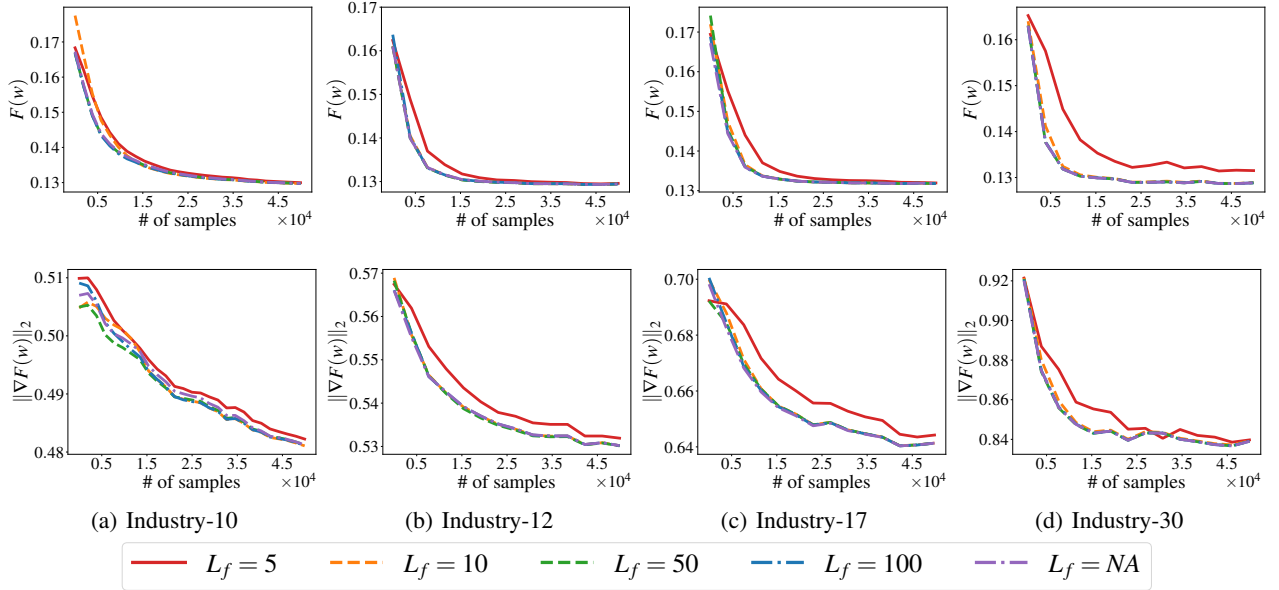


Figure 2. Results for Risk-Averse Portfolio Optimization.

Figure 1 shows the loss value and the norm of the gradient against the number of samples drawn by each method, and all curves are averaged over 20 runs. We can find that our method converges much faster than other algorithms in all tasks. Specifically, both the loss and the gradient of SMVR decrease more quickly, demonstrating the low sample complexity of the proposed method.

We also conduct experiments on tuning the parameter  $L_f$  for the projection operation  $\Pi_{L_f}$ . For theoretical analysis, if  $L_f$

is set bigger than the actual upper bound of the gradient, the convergence rate remains the same order, just with a bigger constant. Here, we tune the  $L_f$  from the set  $\{5, 10, 50, 100\}$ , and the results are shown in Figure 2, where  $L_f = NA$  means the projection is not used (it equals to setting  $L_f$  as an extremely large number, such as  $1e7$ ). We find that the method performs very closely as long as  $L_f$  is set as a large number and would perform worse when  $L_f$  is small. In practical use, we can simply set  $L_f$  as a large number.

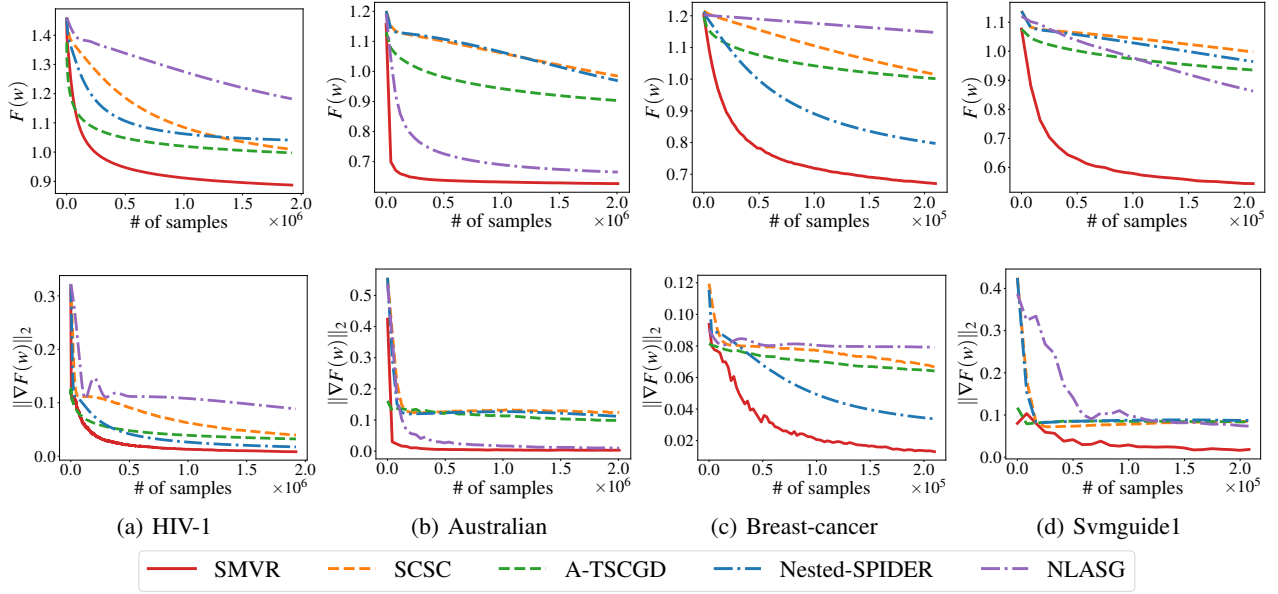


Figure 3. Results for Hierarchical Tilted Empirical Risk Minimization.

Table 2. Classification accuracies (%) for Hierarchical Tilted Empirical Risk Minimization.

| METHOD        | Hiv-1           |                 | AUSTRALIAN SCALE |                 | BREAST-CANCER   |                 | SVMGUIDE1       |                 |
|---------------|-----------------|-----------------|------------------|-----------------|-----------------|-----------------|-----------------|-----------------|
|               | RARE            | OVERALL         | RARE             | OVERALL         | RARE            | OVERALL         | RARE            | OVERALL         |
| A-TSCGD       | 71.3±2.9        | 88.1±1.2        | 62.7±8.9         | 72.5±5.2        | 55.1±8.5        | 80.6±4.4        | 76.5±4.6        | 85.8±2.1        |
| SCSC          | 76.7±2.4        | 88.7±1.2        | 61.8±8.4         | 72.5±5.2        | 55.6±8.0        | 80.7±4.1        | 75.8±4.7        | 85.5±2.1        |
| NESTED-SPIDER | 47.3±9.2        | 63.0±6.7        | 68.5±9.7         | 72.9±4.8        | 61.8±8.6        | 83.3±6.6        | 74.3±5.4        | 84.8±2.4        |
| NLAGS         | 69.9±3.0        | 88.1±1.3        | 79.4±8.4         | 78.7±5.8        | 42.0±8.5        | 76.7±5.2        | 73.3±5.2        | 85.0±2.2        |
| <b>SMVR</b>   | <b>79.0±2.5</b> | <b>90.0±1.2</b> | <b>82.9±8.2</b>  | <b>82.6±4.6</b> | <b>74.0±8.5</b> | <b>87.0±4.3</b> | <b>80.9±3.0</b> | <b>87.3±2.7</b> |

## 5.2. Hierarchical Tilted Empirical Risk Minimization

Hierarchical Tilted Empirical Risk Minimization (TERM) is a method proposed by Li et al. (2021a;b), which can deal with noisy and imbalanced machine learning problems simultaneously. TERM objective is given by  $\tilde{R}(w) := \frac{1}{t} \log \left( \frac{1}{N} \sum_{i \in [N]} e^{t l(w; z_i)} \right)$ , where  $l(w; z_i)$  is the loss on the sample  $z_i$  from data  $\{z_1, \dots, z_N\}$ . It can mitigate outliers with parameter  $t < 0$  and handle class imbalance when  $t > 0$ . When the task involves outliers and class imbalance at the same time, Hierarchical TERM can be used:

$$\tilde{J}(w) := \frac{1}{t} \log \left( \frac{1}{|\mathcal{D}|} \sum_{\mathcal{G} \subseteq \mathcal{D}} |\mathcal{G}| e^{t \tilde{R}_{\mathcal{G}}(w)} \right),$$

$$\text{with } \tilde{R}_{\mathcal{G}}(w) := \frac{1}{\tau} \log \left( \frac{1}{|\mathcal{G}|} \sum_{z \in \mathcal{G}} e^{\tau l(w; z)} \right),$$

where  $\mathcal{D}$  represents all training samples and  $\mathcal{G}$  denotes samples for one specific class. Parameter  $t$  and  $\tau$  are constants dealing with different goals (i.e., outliers and class imbal-

ance). It is a four-level stochastic compositional optimization problem, with each layer represented as:

$$f_1(w) = \frac{1}{|\mathcal{G}|} \sum_{z \in \mathcal{G}} e^{\tau l(w; z)}, \quad f_2(x) = \frac{1}{\tau} \log(x),$$

$$f_3(y) = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{G} \subseteq \mathcal{D}} |\mathcal{G}| e^{t y}, \quad f_4(z) = \frac{1}{t} \log(z).$$

In the experiment, we use the "HIV-1"<sup>3</sup>, "Australian"<sup>4</sup>, "Breast-cancer"<sup>4</sup> and "svmguide1"<sup>4</sup> dataset, and make the training data noisy and imbalanced, where nearly 30% of the labels are reshuffled and the number of rare class versus common class is 1:20. We set  $\tau = -2$ ,  $t = 10$  according to the origin paper and repeat each experiment 20 times. As shown in Figure 3, our SMVR performs best among all algorithms. The loss value and the norm of the gradient converge to a small value more quickly than other methods. We also report the classification accuracy in Table 2. It shows that

<sup>3</sup><https://archive.ics.uci.edu/ml/datasets.php>

<sup>4</sup><https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/>



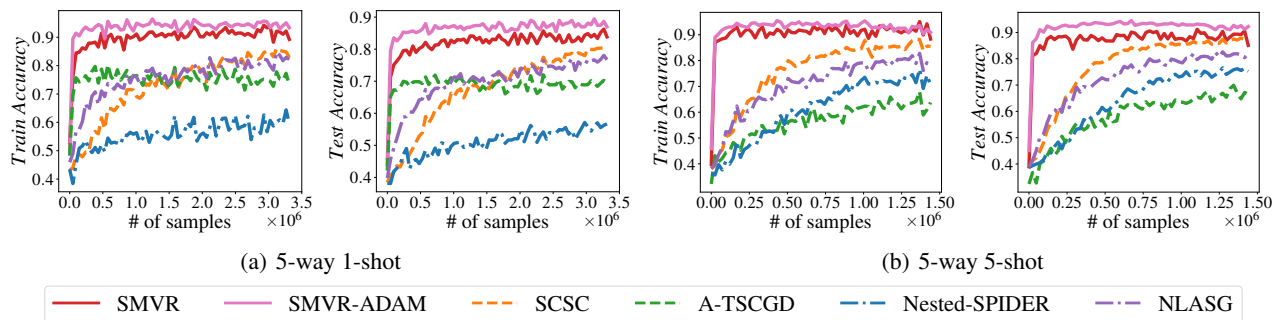


Figure 4. Results for Multi-step Model-Agnostic Meta-Learning.

SMVR achieves the highest accuracy on the rare class and the overall task simultaneously, indicating the effectiveness of our method.

### 5.3. Multi-Step Model-Agnostic Meta-Learning

At last, we conduct experiments on Multi-step Model-Agnostic Meta-Learning (MAML). Multi-step MAML aims to find a good initialization point that performs well in different tasks after taking a few steps of gradient descent. Classical one-step MAML can be formed as:

$$\min_{\mathbf{x}} F(\boldsymbol{\theta}) := \frac{1}{M} \sum_{m=1}^M F_m(\mathbf{x} - \alpha \nabla F_m(\mathbf{x})),$$

with  $F_m(\boldsymbol{\theta}) := \mathbb{E}_{\xi_m} [f(\boldsymbol{\theta}; \xi_m)]$

where  $\alpha$  is the learning rate,  $F_m$  denotes the loss on task  $m$  and  $\xi_m$  represents the training samples for task  $m$ . One-step MAML is a two-level problem since it updates the initial point once and then evaluates on different tasks. In practice, we usually update the initial point more times to achieve better results. For example, Finn et al. (2017) use five-step updates, which is a six-level compositional problem.

Following Finn et al. (2017), we conduct experiments on 5-way 1-shot and 5-shot task on Omniglot dataset (Lake et al., 2011). Each task is a 5-class classification problem, with only 1 or 5 training samples for each class. We conduct 5-step MAML and repeat each experiment 3 times.

We report the accuracy of different methods against the number of training samples in Figure 4. Since adaptive learning rates are widely used in neural networks, which are also applied in Multi-step MAML, we implement Adaptive SMVR methods in this task, denoted as SMVR-ADAM. We use the adaptive learning rate defined in (4) and (5) and choose the commonly used Adam-type. As can be seen, the accuracy of SMVR and SMVR-ADAM increases rapidly both in training sets and testing sets, and outperforms other methods dramatically. Although SMVR and SMVR-ADAM enjoy the same sample complexity, the latter converges faster in practice due to the adaptive learning rate used.

## 6. Conclusion

In this paper, we propose an optimal algorithm named SMVR for stochastic multi-level composition optimization. We prove that the proposed algorithm, by using variance reduced estimator of function values and Jacobians, can achieve the sample complexity of  $\mathcal{O}(1/\epsilon^3)$  for finding an  $\epsilon$ -stationary point. This complexity matches the lower bound even in the one-level setting, and our method avoids using batches in any iterations. When the objective function further satisfies the convexity or PL condition, we develop a stage-wise version of SMVR to obtain the optimal complexity of  $\mathcal{O}(1/\epsilon^2)$  or  $\mathcal{O}(1/\epsilon)$ . To take advantage of adaptive learning rates, we also propose Adaptive SMVR, which can achieve the same complexity with the learning rate changing adaptively. Experiments on three real-world tasks demonstrate the superiority of the proposed method.

## Acknowledgements

W. Jiang, Y. Wang and L. Zhang were partially supported by NSFC (62122037, 61921006). B. Wang and T. Yang were partially supported by NSF Grant 2110545, NSF Career Award 1844403. The authors would like to thank the anonymous reviewers for their helpful comments.

## References

- Agarwal, A., Bartlett, P. L., Ravikumar, P., and Wainwright, M. J. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *IEEE Transactions on Information Theory*, 58(5):3235–3249, 2012.
- Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. E. Lower bounds for non-convex stochastic optimization. *ArXiv e-prints*, arXiv:1912.02365, 2019.
- Balasubramanian, K., Ghadimi, S., and Nguyen, A. Stochastic multi-level composition optimization algorithms with

- level-independent convergence rates. *ArXiv e-prints*, arXiv:2008.10526, 2021.
- Bruno, S., Ahmed, S., Shapiro, A., and Street, A. Risk neutral and risk averse approaches to multistage renewable investment planning under uncertainty. *European Journal of Operational Research*, 250(3):979–989, 2016.
- Charles, Z. and Papailiopoulos, D. Stability and generalization of learning algorithms that converge to global optima. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 745–754, 2018.
- Chen, T., Sun, Y., and Yin, W. Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. *IEEE Transactions on Signal Processing*, 69:4937–4948, 2021.
- Chewi, S., Maunu, T., Rigollet, P., and Stromme, A. Gradient descent algorithms for Bures-Wasserstein barycenters. In *Proceedings of the 33rd Conference on Learning Theory*, pp. 1276–1304, 2020.
- Cole, S., Giné, X., and Vickery, J. How Does Risk Management Influence Production Decisions? Evidence from a Field Experiment. *The Review of Financial Studies*, 30(6):1935–1970, 2017.
- Cutkosky, A. and Orabona, F. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems 32*, pp. 15210–15219, 2019.
- Dann, C., Neumann, G., and Peters, J. Policy evaluation with temporal differences: a survey and comparison. *Journal of Machine Learning Research*, 15:809–883, 2014.
- Dentcheva, D., Penev, S. I., and Ruszczyński, A. Statistical estimation of composite risk functionals and risk optimization problems. *Annals of the Institute of Statistical Mathematics*, 69(4):737–760, 2017.
- Duchi, J., Hazan, E., and Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pp. 257–269, 2010.
- Fang, C., Li, C. J., Lin, Z., and Zhang, T. Spider: Near-optimal non-convex optimization via stochastic path integrated differential estimator. *ArXiv e-prints*, arXiv:1807.01695, 2018.
- Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 1126–1135, 2017.
- Ghadimi, S., Ruszczyński, A., and Wang, M. A single timescale stochastic approximation method for nested stochastic optimization. *SIAM Journal on Optimization*, 30(1):960–979, 2020.
- Guo, Z., Xu, Y., Yin, W., Jin, R., and Yang, T. On stochastic moving-average estimators for non-convex optimization. *ArXiv e-prints*, arXiv:2104.14840, 2021.
- Huo, Z., Gu, B., Liu, J., and Huang, H. Accelerated method for stochastic composition optimization with nonsmooth regularization. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 3287–3294, 2018.
- Ji, K., Yang, J., and Liang, Y. Multi-step model-agnostic meta-learning: Convergence and improved algorithms. *ArXiv e-prints*, arXiv:2002.07836, 2020.
- Karimi, H., Nutini, J., and Schmidt, M. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases*, pp. 795–811, 2016.
- Kingma, D. P. and Ba, J. L. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.
- Lake, B. M., Salakhutdinov, R., Gross, J., and Tenenbaum, J. B. One shot learning of simple visual concepts. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 33, 2011.
- Li, T., Beirami, A., Sanjabi, M., and Smith, V. Tilted empirical risk minimization. In *International Conference on Learning Representations*, 2021a.
- Li, T., Beirami, A., Sanjabi, M., and Smith, V. On tilted losses in machine learning: Theory and applications. *ArXiv e-prints*, arXiv:2109.06141, 2021b.
- Liu, L., Liu, J., and Tao, D. Variance reduced methods for non-convex composition optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021.
- Luo, L., Xiong, Y., and Liu, Y. Adaptive gradient methods with dynamic bound of learning rate. In *International Conference on Learning Representations*, 2019.
- Nguyen, L. M., Liu, J., Scheinberg, K., and Takác, M. Sarah: A novel method for machine learning problems using stochastic recursive gradient. *ArXiv e-prints*, arXiv:1703.00102, 2017.
- Nouiehed, M., Sanjabi, M., Huang, T., Lee, J., and Razaviyayn, M. Solving a class of non-convex min-max games using iterative first order methods. In *Advances in Neural*

- Information Processing Systems 33*, pp. 14905–14916, 2019.
- Qi, Q., Guo, Z., Xu, Y., Jin, R., and Yang, T. An on-line method for a class of distributionally robust optimization with non-convex objectives. *ArXiv e-prints*, arXiv:2006.10138, 2021.
- Reddi, S. J., Kale, S., and Kumar, S. On the convergence of Adam and beyond. In *International Conference on Learning Representations*, 2018.
- Shapiro, A., Dentcheva, D., and Ruszczyński, A. *Lectures on Stochastic Programming: Modeling and Theory, Third Edition*. Society for Industrial and Applied Mathematics, 2021.
- Wang, M., Fang, E. X., and Liu, H. Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. *Mathematical Programming*, 161(1-2):419–449, 2017a.
- Wang, M., Liu, J., and Fang, E. X. Accelerating stochastic composition optimization. *Journal of Machine Learning Research*, 18:105:1–105:23, 2017b.
- Wang, Z., Ji, K., Zhou, Y., Liang, Y., and Tarokh, V. Spiderboost: A class of faster variance-reduced algorithms for nonconvex optimization. *ArXiv e-prints*, arXiv:1810.10690, 2018.
- Xie, Y., Wu, X., and Ward, R. A. Linear convergence of adaptive stochastic gradient descent. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, pp. 1475–1485, 2020.
- Yang, S., Wang, M., and Fang, E. X. Multilevel stochastic gradient methods for nested composition optimization. *SIAM Journal on Optimization*, 29(1):616–659, 2019.
- Yuan, H. and Hu, W. Stochastic recursive momentum method for non-convex compositional optimization. *ArXiv e-prints*, arXiv:2006.01688, 2020.
- Yuan, H., Lian, X., Li, C. J., Liu, J., and Hu, W. Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent. In *Advances in Neural Information Processing Systems 33*, pp. 14905–14916, 2019.
- Zhang, J. and Xiao, L. A composite randomized incremental gradient method. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 7454–7462, 2019a.
- Zhang, J. and Xiao, L. A stochastic composite gradient method with incremental variance reduction. In *Advances in Neural Information Processing Systems 33*, pp. 9075–9085, 2019b.
- Zhang, J. and Xiao, L. Multilevel composite stochastic optimization via nested variance reduction. *SIAM Journal on Optimization*, 31(2):1131–1157, 2021.
- Zhang, Z. and Lan, G. Optimal algorithms for convex nested stochastic composite optimization. *ArXiv e-prints*, arXiv:2011.10076, 2021.

## A. Proof of Theorem 1

We first provide some supporting lemmas and then conclude to show the sample complexity of the proposed method.

**Lemma 2.** *The objective function  $F$  is  $L_F$ -smooth, where  $L_F := L_f^{2K-1} L_J \sum_{i=1}^K \frac{1}{L_f^i}$ .*

*Proof.* Denote  $F_i(x) = f_i \circ f_{i-1} \circ \dots \circ f_1(x)$ . We know  $F_i(x)$  is  $L_f^i$  Lipschitz continuous, since when  $i \geq 2$ :

$$\begin{aligned} \|F_i(x) - F_i(y)\| &= \|f_i(F_{i-1}(x)) - f_i(F_{i-1}(y))\| \\ &\leq L_f \|F_{i-1}(x) - F_{i-1}(y)\| \\ &\leq L_f^{i-1} \|F_1(x) - F_1(y)\| \\ &\leq L_f^i \|x - y\| \end{aligned}$$

Then, we have:

$$\begin{aligned} &\|\nabla F_K(x) - \nabla F_K(y)\| \\ &= \|\nabla f_K(F_{K-1}(x)) \nabla F_{K-1}(x) - \nabla f_K(F_{K-1}(y)) \nabla F_{K-1}(y)\| \\ &= \|\nabla f_K(F_{K-1}(x)) (\nabla F_{K-1}(x) - \nabla F_{K-1}(y)) + \nabla F_{K-1}(y) (\nabla f_K(F_{K-1}(x)) - \nabla f_K(F_{K-1}(y)))\| \\ &\leq \|\nabla f_K(F_{K-1}(x))\| \|\nabla F_{K-1}(x) - \nabla F_{K-1}(y)\| + \|\nabla F_{K-1}(y)\| \|\nabla f_K(F_{K-1}(x)) - \nabla f_K(F_{K-1}(y))\| \\ &\leq L_f \|\nabla F_{K-1}(x) - \nabla F_{K-1}(y)\| + L_f^{K-1} L_J \|F_{K-1}(x) - F_{K-1}(y)\| \\ &\leq L_f \|\nabla F_{K-1}(x) - \nabla F_{K-1}(y)\| + L_f^{2K-2} L_J \|x - y\| \\ &\leq L_f^2 \|\nabla F_{K-2}(x) - \nabla F_{K-2}(y)\| + (L_f^{2K-3} L_J + L_f^{2K-2} L_J) \|x - y\| \\ &\leq L_f^{K-1} \|\nabla F_1(x) - \nabla F_1(y)\| + L_f^{2K-1} L_J \sum_{i=1}^{K-1} \frac{1}{L_f^i} \|x - y\| \\ &\leq L_f^{2K-1} \frac{1}{L_f^K} L_J \|x - y\| + L_f^{2K-1} L_J \sum_{i=1}^{K-1} \frac{1}{L_f^i} \|x - y\| \\ &= L_f^{2K-1} L_J \sum_{i=1}^K \frac{1}{L_f^i} \|x - y\| \end{aligned}$$

□

A similar result can also be found in Lemma 2.1 of (Balasubramanian et al., 2021). This property is unsurprising since the composition of two smooth and Lipschitz functions is still smooth and Lipschitz.

**Lemma 3.** *Let  $\eta_t \leq \frac{1}{2L_F}$ , we have following guarantee:*

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t}{4} \|\mathbf{v}_t\|^2.$$

*Proof.* Due to the smoothness of function  $F$  and the definition of  $\mathbf{w}_{t+1}$ , we have:

$$\begin{aligned} F(\mathbf{w}_{t+1}) &\leq F(\mathbf{w}_t) + \langle \nabla F(\mathbf{w}_t), \mathbf{w}_{t+1} - \mathbf{w}_t \rangle + \frac{L_F}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 \\ &= F(\mathbf{w}_t) - \eta_t \langle \nabla F(\mathbf{w}_t), \mathbf{v}_t \rangle + \frac{\eta_t^2 L_F}{2} \|\mathbf{v}_t\|^2 \\ &\leq F(\mathbf{w}_t) - \eta_t \langle \nabla F(\mathbf{w}_t), \mathbf{v}_t \rangle + \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta_t}{2} \|\mathbf{v}_t\|^2 - \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t}{2} \|\mathbf{v}_t\|^2 + \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 \\ &= F(\mathbf{w}_t) + \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 \end{aligned}$$

□

We can derive the term  $\|\nabla F(\mathbf{w}_t)\|^2$  from the right side. Since  $F(\mathbf{w}_{t+1})$  and  $F(\mathbf{w}_t)$  can be eliminated when summing up, the only thing need to show is that the estimated gradient  $\mathbf{v}_t$  is not too far away from the true gradient  $\nabla F(\mathbf{w}_t)$ . We prove this in the following lemmas.

**Lemma 4.** Denote that  $\nabla \widehat{F}_K(\mathbf{w}_t) := \prod_{i=1}^K \nabla f_i(\mathbf{u}_t^{i-1})$  and  $C_i := L_f^{K-1} L_J \sum_{j=1}^{K-i-1} L_f^j$ . For  $K \geq 2$ , we have:

$$\left\| \nabla F(\mathbf{w}_t) - \nabla \widehat{F}_K(\mathbf{w}_t) \right\|^2 \leq K \sum_{i=1}^{K-1} C_i^2 \|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\|^2.$$

*Proof.* First, we define that  $\mathbf{y}_t^i := f_i \circ f_{i-1} \circ \dots \circ f_1(\mathbf{w}_t)$ ,  $\nabla \widehat{F}_i(\mathbf{w}_t) := \nabla f_1(\mathbf{w}_t) \dots \nabla f_i(\mathbf{u}_t^{i-1})$ .

$$\begin{aligned} \left\| \nabla F_1(\mathbf{w}_t) - \nabla \widehat{F}_1(\mathbf{w}_t) \right\| &= 0, \\ \left\| \nabla F_2(\mathbf{w}_t) - \nabla \widehat{F}_2(\mathbf{w}_t) \right\| &= \left\| \nabla f_1(\mathbf{w}_t) \nabla f_2(\mathbf{y}_t^1) - \nabla f_1(\mathbf{w}_t) \nabla f_2(\mathbf{u}_t^1) \right\| \\ &\leq L_f L_J \|\mathbf{y}_t^1 - \mathbf{u}_t^1\|, \\ \left\| \nabla F_3(\mathbf{w}_t) - \nabla \widehat{F}_3(\mathbf{w}_t) \right\| &= \left\| \nabla f_1(\mathbf{w}_t) \nabla f_2(\mathbf{y}_t^1) \nabla f_3(\mathbf{y}_t^2) - \nabla f_1(\mathbf{w}_t) \nabla f_2(\mathbf{u}_t^1) \nabla f_3(\mathbf{u}_t^2) \right\| \\ &\leq L_f^2 L_J (\|\mathbf{y}_t^2 - \mathbf{u}_t^2\| + \|\mathbf{y}_t^1 - \mathbf{u}_t^1\|), \\ &\dots \\ \left\| \nabla F_K(\mathbf{w}_t) - \nabla \widehat{F}_K(\mathbf{w}_t) \right\| &\leq L_f^{K-1} L_J \sum_{i=1}^{K-1} \|\mathbf{y}_t^i - \mathbf{u}_t^i\|, \end{aligned}$$

Besides, we also have

$$\begin{aligned} \|\mathbf{y}_t^2 - \mathbf{u}_t^2\| &= \|f_2 \circ f_1(\mathbf{w}_t) - \mathbf{u}_t^2\| \leq \|f_2 \circ f_1(\mathbf{w}_t) - f_2(\mathbf{u}_t^1)\| + \|f_2(\mathbf{u}_t^1) - \mathbf{u}_t^2\| \\ &\leq L_f \|f_1(\mathbf{w}_t) - \mathbf{u}_t^1\| + \|f_2(\mathbf{u}_t^1) - \mathbf{u}_t^2\|, \\ \|\mathbf{y}_t^3 - \mathbf{u}_t^3\| &= \|f_3 \circ f_2 \circ f_1(\mathbf{w}_t) - \mathbf{u}_t^3\| \leq \|f_3 \circ f_2 \circ f_1(\mathbf{w}_t) - f_3(\mathbf{u}_t^2)\| + \|f_3(\mathbf{u}_t^2) - \mathbf{u}_t^3\| \\ &\leq L_f \|\mathbf{y}_t^2 - \mathbf{u}_t^2\| + \|f_3(\mathbf{u}_t^2) - \mathbf{u}_t^3\| \\ &\leq L_f (L_f \|f_1(\mathbf{w}_t) - \mathbf{u}_t^1\| + \|f_2(\mathbf{u}_t^1) - \mathbf{u}_t^2\|) + \|f_3(\mathbf{u}_t^2) - \mathbf{u}_t^3\| \\ &\dots \\ \|\mathbf{y}_t^i - \mathbf{u}_t^i\| &\leq L_f \|\mathbf{y}_t^{i-1} - \mathbf{u}_t^{i-1}\| + \|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\| \\ &\leq \sum_{j=1}^i L_f^{i-j} \|f_j(\mathbf{u}_t^{j-1}) - \mathbf{u}_t^j\| \end{aligned}$$

To this end, we can conclude that:

$$\left\| \nabla F(\mathbf{w}_t) - \nabla \widehat{F}_K(\mathbf{w}_t) \right\| \leq \sum_{i=1}^{K-1} C_i \|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\|,$$

where  $C_i := L_f^{K-1} L_J (1 + L_f + \dots + L_f^{K-i-1})$ . □

**Lemma 5.** The estimated error of gradient can be bounded as:

$$\mathbb{E} \left[ \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 \right] \leq 2K L_f^{2(K-1)} \sum_{i=1}^K \mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] + 2K L_f^2 \sum_{i=1}^{K-1} \mathbb{E} \left[ \|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\|^2 \right].$$

*Proof.* Consider the definition of  $\mathbf{v}_t$ :

$$\mathbb{E} \left[ \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2 \right] \leq 2 \mathbb{E} \left[ \underbrace{\left\| \prod_{i=1}^K \mathbf{v}_t^i - \prod_{i=1}^K \nabla f_i(\mathbf{u}_t^{i-1}) \right\|^2}_{:=\diamond} \right] + 2 \mathbb{E} \left[ \underbrace{\left\| \nabla \widehat{F}_K(\mathbf{w}_t) - \nabla F(\mathbf{w}_t) \right\|^2}_{:=\clubsuit} \right].$$

Based on Lemma 4, we have:

$$\clubsuit \leq K \sum_{i=1}^{K-1} C_i^2 \mathbb{E} \left[ \|\mathbf{f}_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\|^2 \right],$$

where  $C_i := L_f^{K-1} L_J \sum_{j=1}^{K-i-1} L_f^j \leq L_F$ .

The first term  $\diamond$  can be upper bounded as:

$$\begin{aligned} \diamond &= \mathbb{E} \left[ \left\| \prod_{i=1}^K \mathbf{v}_t^i - \prod_{i=1}^K \nabla f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ &\leq K \mathbb{E} \left[ \left\| \prod_{i=1}^K \nabla f_i(\mathbf{u}_t^{i-1}) - \mathbf{v}_t^1 \prod_{i=2}^K \nabla f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ &\quad + K \mathbb{E} \left[ \left\| \mathbf{v}_t^1 \prod_{i=2}^K \nabla f_i(\mathbf{u}_t^{i-1}) - \mathbf{v}_t^1 \mathbf{v}_t^2 \prod_{i=3}^K \nabla f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ &\quad + \dots \\ &\quad + K \mathbb{E} \left[ \left\| \left( \prod_{i=1}^{K-1} \mathbf{v}_t^i \right) \nabla f_K(\mathbf{u}_t^{K-1}) - \prod_{i=1}^K \mathbf{v}_t^i \right\|^2 \right] \\ &\leq K \left( \sum_{i=1}^K L_f^{2(K-1)} \mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] \right). \end{aligned}$$

□

**Lemma 6.** *The variance of the estimated gradient and function value satisfies the following guarantee:*

$$\begin{aligned} \mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] &\leq (1 - \beta_t) \mathbb{E} \left[ \|\mathbf{v}_{t-1}^i - \nabla f_i(\mathbf{u}_{t-1}^{i-1})\|^2 \right] + 2\beta_t^2 \sigma_J^2 + 2\mathcal{L}_J^2 \mathbb{E} \left[ \|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\|^2 \right] \\ \mathbb{E} \left[ \|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\|^2 \right] &\leq (1 - \beta_t) \mathbb{E} \left[ \|\mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1})\|^2 \right] + 2\beta_t^2 \sigma_f^2 + 2\mathcal{L}_f^2 \mathbb{E} \left[ \|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\|^2 \right] \end{aligned}$$

*Proof.* Consider the update  $\mathbf{v}_t^i = \Pi_{L_f} \left[ (1 - \beta_t) \mathbf{v}_{t-1}^i + \beta_t \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i) + (1 - \beta_t) (\nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i)) \right]$  and  $\Pi_{L_f}[\nabla f_i(\mathbf{u}_t^{i-1})] = \nabla f_i(\mathbf{u}_t^{i-1})$ .

$$\begin{aligned} &\mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] \\ &= \mathbb{E} \left[ \left\| \Pi_{L_f} \left[ (1 - \beta_t) \mathbf{v}_{t-1}^i + \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - (1 - \beta_t) \nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) \right] - \Pi_{L_f}[\nabla f_i(\mathbf{u}_t^{i-1})] \right\|^2 \right] \\ &\leq \mathbb{E} \left[ \left\| (1 - \beta_t) \mathbf{v}_{t-1}^i + \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - (1 - \beta_t) \nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ &= \mathbb{E} \left[ \left\| (1 - \beta_t) (\mathbf{v}_{t-1}^i - \nabla f_i(\mathbf{u}_{t-1}^{i-1})) + (\nabla f_i(\mathbf{u}_{t-1}^{i-1}) - \nabla f_i(\mathbf{u}_t^{i-1}) - (\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i))) \right. \right. \\ &\quad \left. \left. + \beta_t (\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1})) \right\|^2 \right] \end{aligned}$$

Note that  $\mathbb{E} \left[ \nabla f_i(\mathbf{u}_{t-1}^{i-1}) - \nabla f_i(\mathbf{u}_t^{i-1}) - (\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i)) + \beta_t (\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1})) \right] = 0$ .

$$\begin{aligned} \mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] &\leq (1 - \beta_t) \mathbb{E} \left[ \|\mathbf{v}_{t-1}^i - \nabla f_i(\mathbf{u}_{t-1}^{i-1})\|^2 \right] + 2\beta_t^2 \mathbb{E} \left[ \|\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] \\ &\quad + 2\mathbb{E} \left[ \|\nabla f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - \nabla f_i(\mathbf{u}_t^{i-1}; \xi_t^i)\|^2 \right] \\ &\leq (1 - \beta_t) \mathbb{E} \left[ \|\mathbf{v}_{t-1}^i - \nabla f_i(\mathbf{u}_{t-1}^{i-1})\|^2 \right] + 2\beta_t^2 \sigma_J^2 + 2\mathcal{L}_J^2 \mathbb{E} \left[ \|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\|^2 \right] \end{aligned}$$

Similarly, consider  $\mathbf{u}_t^i = (1 - \beta_t)\mathbf{u}_{t-1}^i + \beta_t f_i(\mathbf{u}_t^{i-1}; \xi_t^i) + (1 - \beta_t)(f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i))$ :

$$\begin{aligned} & \mathbb{E} \left[ \left\| f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i \right\|^2 \right] \\ &= \mathbb{E} \left[ \left\| (1 - \beta_t)\mathbf{u}_{t-1}^i + f_i(\mathbf{u}_t^{i-1}; \xi_t^i) - (1 - \beta_t)f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ &= \mathbb{E} \left[ \left\| (1 - \beta_t)(\mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1})) + (f_i(\mathbf{u}_{t-1}^{i-1}) - f_i(\mathbf{u}_t^{i-1}) - (f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_t^{i-1}; \xi_t^i))) \right. \right. \\ & \quad \left. \left. + \beta_t (f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_{t-1}^{i-1})) \right\|^2 \right] \end{aligned}$$

Note that  $\mathbb{E} [f_i(\mathbf{u}_{t-1}^{i-1}) - f_i(\mathbf{u}_t^{i-1}) - (f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_t^{i-1}; \xi_t^i)) + \beta_t (f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_{t-1}^{i-1}))] = 0$

$$\begin{aligned} & \mathbb{E} \left[ \left\| \mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \\ & \leq (1 - \beta_t) \mathbb{E} \left\| \mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1}) \right\|^2 + 2\beta_t^2 \mathbb{E} \left[ \left\| f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_{t-1}^{i-1}) \right\|^2 \right] + 2\mathbb{E} \left[ \left\| f_i(\mathbf{u}_{t-1}^{i-1}; \xi_t^i) - f_i(\mathbf{u}_t^{i-1}; \xi_t^i) \right\|^2 \right] \\ & \leq (1 - \beta_t) \mathbb{E} \left\| \mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1}) \right\|^2 + 2\beta_t^2 \sigma_f^2 + 2\mathcal{L}_f^2 \mathbb{E} \left[ \left\| \mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1} \right\|^2 \right] \end{aligned}$$

□

**Lemma 7.** *We have the following guarantee:*

$$\sum_{i=1}^K \mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1} \right\|^2 \right] \leq \left( \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \right) \left( \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 K + 2\beta_{t+1}^2 K \sum_{i=1}^K \mathbb{E} \left[ \left\| \mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right] \right)$$

*Proof.* First, we discuss two cases:

1. ( $i = 1$ ):  $\mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1} \right\|^2 \right] = \mathbb{E} \left[ \left\| \mathbf{w}_{t+1} - \mathbf{w}_t \right\|^2 \right] = \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right]$ .
2. ( $2 \leq i \leq K$ ):

$$\begin{aligned} & \mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1} \right\|^2 \right] \\ &= \mathbb{E} \left[ \left\| \beta_{t+1} (f_{i-1}(\mathbf{u}_t^{i-2}) - \mathbf{u}_t^{i-1}) + (f_{i-1}(\mathbf{u}_{t+1}^{i-2}; \xi_{t+1}^{i-1}) - f_{i-1}(\mathbf{u}_t^{i-2}; \xi_{t+1}^{i-1})) \right. \right. \\ & \quad \left. \left. + \beta_{t+1} (f_{i-1}(\mathbf{u}_t^{i-2}; \xi_{t+1}^{i-1}) - f_{i-1}(\mathbf{u}_t^{i-2})) \right\|^2 \right] \\ & \leq 2\mathbb{E} \left[ \left\| \beta_{t+1} (f_{i-1}(\mathbf{u}_t^{i-2}) - \mathbf{u}_t^{i-1}) + \beta_{t+1} (f_{i-1}(\mathbf{u}_t^{i-2}; \xi_{t+1}^{i-1}) - f_{i-1}(\mathbf{u}_t^{i-2})) \right\|^2 \right] + 2\mathcal{L}_f^2 \mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-2} - \mathbf{u}_t^{i-2} \right\|^2 \right] \\ & \leq 2\beta_{t+1}^2 \left\| f_{i-1}(\mathbf{u}_t^{i-2}) - \mathbf{u}_t^{i-1} \right\|^2 + 2\beta_{t+1}^2 \sigma_f^2 + 2\mathcal{L}_f^2 \mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-2} - \mathbf{u}_t^{i-2} \right\|^2 \right]. \end{aligned}$$

Denote  $\Upsilon_t^i := \mathbb{E} \left[ \left\| \mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1}) \right\|^2 \right]$  and  $\mathcal{A}^i := \mathbb{E} \left[ \left\| \mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1} \right\|^2 \right]$ , we have  $A^i \leq 2\mathcal{L}_f^2 A^{i-1} + 2\beta_{t+1}^2 \Upsilon_t^{i-1} + 2\beta_{t+1}^2 \sigma_f^2$  for  $i \geq 2$ . Then we can get:

$$\begin{aligned} A^1 & \leq \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] \\ A^2 & \leq (2\mathcal{L}_f^2) \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 + 2\beta_{t+1}^2 \Upsilon_t^1 \\ A^3 & \leq (2\mathcal{L}_f^2)^2 \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 (1 + 2\mathcal{L}_f^2) + 2\beta_{t+1}^2 (2\mathcal{L}_f^2 \Upsilon_t^1 + \Upsilon_t^2) \\ & \dots \\ A^i & \leq (2\mathcal{L}_f^2)^{i-1} \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 \sum_{j=1}^{i-1} (2\mathcal{L}_f^2)^{j-1} + 2\beta_{t+1}^2 \sum_{j=1}^{i-1} (2\mathcal{L}_f^2)^{i-1-j} \Upsilon_t^j \\ & \leq (2\mathcal{L}_f^2)^{i-1} \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 \sum_{j=1}^K (2\mathcal{L}_f^2)^{j-1} + 2\beta_{t+1}^2 \sum_{j=1}^K (2\mathcal{L}_f^2)^{K-j} \Upsilon_t^j \end{aligned}$$

When summing up, we have:

$$\begin{aligned} \sum_{i=1}^K A^i &\leq \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 K \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} + 2\beta_{t+1}^2 K \sum_{i=1}^K (2\mathcal{L}_f^2)^{K-i} \Upsilon_t^i \\ &\leq \left( \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \right) \left( \mathbb{E} \left[ \eta_t^2 \|\mathbf{v}_t\|^2 \right] + 2\beta_{t+1}^2 \sigma_f^2 K + 2\beta_{t+1}^2 K \sum_{i=1}^K \Upsilon_t^i \right), \end{aligned}$$

□

Now we finish the proof of Theorem 1. Denote that  $\Gamma_t = F(\mathbf{w}_t) + \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i)$ ,  $\Phi_t^i := \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2$  and  $\Upsilon_t^i := \mathbb{E} \left[ \|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\|^2 \right]$ . Based on Lemma 3, we have:

$$\begin{aligned} &\mathbb{E} [\Gamma_{t+1} - \Gamma_t] \\ &= \mathbb{E} \left[ F(\mathbf{w}_{t+1}) - F(\mathbf{w}_t) + \frac{1}{c_0 \eta_t} \sum_{i=1}^K (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i) \right] \\ &\leq \mathbb{E} \left[ \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 + \frac{1}{c_0 \eta_t} \sum_{i=1}^K (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i) \right] \end{aligned}$$

Define constant  $L_1 = \max \left\{ 1, KL_f^{2(K-1)}, KL_F^2, 2K(\sigma_J^2 + \sigma_f^2), 2(L_J^2 + L_f^2)(1 + 2K + 2K\sigma_f^2) \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \right\}$ .

By summing up and rearranging, we have:

$$\begin{aligned} &\mathbb{E} \left[ \sum_{t=1}^T \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 \right] \\ &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \sum_{t=1}^T \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 - \sum_{t=1}^T \sum_{i=1}^K \frac{1}{c_0 \eta_{t-1}} (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. + \sum_{t=1}^T \sum_{i=1}^K \frac{1}{c_0 \eta_t} (\Phi_{t+1}^i + \Upsilon_{t+1}^i) \right] \\ &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + L_1 \sum_{t=1}^T \sum_{i=1}^K \eta_t (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 - \sum_{t=1}^T \sum_{i=1}^K \frac{1}{c_0 \eta_{t-1}} (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. + \sum_{t=1}^T \sum_{i=1}^K \frac{(1 - \beta_{t+1})}{c_0 \eta_t} (\Phi_t^i + \Upsilon_t^i) + \sum_{t=1}^T \frac{2\beta_{t+1}^2 K (\sigma_J^2 + \sigma_f^2)}{c_0 \eta_t} + \sum_{t=1}^T \sum_{i=1}^K \frac{2(\mathcal{L}_J^2 + \mathcal{L}_f^2)}{c_0 \eta_t} \|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\|^2 \right] \\ &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + L_1 \sum_{t=1}^T \sum_{i=1}^K \eta_t (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \frac{\eta_t}{4} \|\mathbf{v}_t\|^2 - \sum_{t=1}^T \sum_{i=1}^K \left( \frac{1}{c_0 \eta_{t-1}} - \frac{1 - \beta_{t+1}}{c_0 \eta_t} \right) (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. + \sum_{t=1}^T \frac{2\beta_{t+1}^2 K (\sigma_J^2 + \sigma_f^2)}{c_0 \eta_t} + \sum_{t=1}^T \frac{L_1}{c_0} \eta_t \|\mathbf{v}_t\|^2 + \sum_{t=1}^T \frac{L_1 \beta_{t+1}^2}{c_0 \eta_t} + \sum_{t=1}^T \sum_{i=1}^K \frac{L_1 \beta_{t+1}^2}{c_0 \eta_t} \Upsilon_t^i \right] \\ &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \sum_{i=1}^K \left( L_1 \eta_t - \frac{1}{c_0 \eta_{t-1}} + \frac{L_1 \beta_{t+1}^2}{c_0 \eta_t} + \frac{(1 - \beta_{t+1})}{c_0 \eta_t} \right) (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. - \sum_{t=1}^T \left( \frac{1}{4} - \frac{L_1}{c_0} \right) \eta_t \|\mathbf{v}_t\|^2 + \sum_{t=1}^T \frac{2L_1 \beta_{t+1}^2}{c_0 \eta_t} \right] \end{aligned}$$



Set  $c_0 = 4L_1$ ,  $\eta_t = (a+t)^{-1/3}$  and  $c = 10L_1^2$ ,  $a = (20L_1^3)^{3/2}$ . Since  $(x+y)^{1/3} - x^{1/3} \leq yx^{-2/3}/3$  and  $a \geq 2$ , we have:

$$\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} = (a+t)^{1/3} - (a+(t-1))^{1/3} \leq \frac{1}{3(a+(t-1))^{2/3}} \leq \frac{1}{3(a/2+t)^{2/3}} \leq \frac{2^{2/3}}{3(a+t)^{2/3}} \leq \eta_t^2$$

Also, we have  $\beta_{t+1} = c\eta_t^2 \leq c\eta_0^2 \leq 10L_1^2a^{-2/3} \leq (2L_1)^{-1}$ , so:

$$L_1\eta_t - \frac{1}{c_0\eta_{t-1}} + \frac{L_1\beta_{t+1}^2}{c_0\eta_t} + \frac{(1-\beta_{t+1})}{c_0\eta_t} \leq L_1\eta_t + \frac{\eta_t^2}{c_0} - \frac{\beta_{t+1}}{2c_0\eta_t} \leq L_1\eta_t + \frac{\eta_t^2}{4L_1} - \frac{5L_1\eta_t}{4} \leq 0$$

So, we have:

$$\begin{aligned} \mathbb{E} \left[ \sum_{t=1}^T \frac{\eta_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 \right] &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \frac{2L_1\beta_{t+1}^2}{c_0\eta_t} \right] \\ &\leq \mathbb{E} \left[ F(\mathbf{w}_1) - F_* + \frac{1}{c_0\eta_0} \sum_{i=1}^K (\Phi_1^i + \Upsilon_1^i) + \sum_{t=1}^T \frac{2L_1\beta_{t+1}^2}{c_0\eta_t} \right] \\ &\leq \mathbb{E} \left[ \Delta_F + \frac{K(\sigma_f^2 + \sigma_J^2)}{c_0\eta_0} + 50L_1^4 \sum_{t=1}^T \eta_t^3 \right] \\ &\leq \Delta_F + \frac{K(\sigma_f^2 + \sigma_J^2)}{c_0\eta_0} + 50L_1^4 \ln(T+1) \end{aligned}$$

The last inequality holds because  $\eta_t^3 \leq (a+t)^{-1} \leq (t+1)^{-1}$  and  $\sum_{t=1}^T (t+1)^{-1} \leq \ln(T+1)$ . Since  $\eta_t$  is decreasing, we have:

$$\mathbb{E} \left[ \eta_T \sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2 \right] \leq 2\Delta_F + \frac{2K(\sigma_f^2 + \sigma_J^2)}{c_0\eta_0} + 100L_1^4 \ln(T+1).$$

Similar to the proof of Theorem 1 in STORM (Cutkosky & Orabona, 2019), denote  $M = 2\Delta_F + 2K(\sigma_f^2 + \sigma_J^2) / (c_0\eta_0) + 100L_1^4 \ln(T+1)$ . Using Cauchy-Schwarz inequality, we have:

$$\mathbb{E} \left[ \sqrt{\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2} \right]^2 \leq \mathbb{E}[1/\eta_T] \mathbb{E} \left[ \eta_T \sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2 \right] \leq \mathbb{E} \left[ \frac{M}{\eta_T} \right] \leq \mathbb{E} \left[ M(a+T)^{1/3} \right],$$

which indicate that

$$\mathbb{E} \left[ \sqrt{\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2} \right] \leq \sqrt{M} (a+T)^{1/6}.$$

Finally, using Cauchy-Schwarz we have  $\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\| / T \leq \sqrt{\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2} / \sqrt{T}$  so that:

$$\mathbb{E} \left[ \frac{\sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|}{T} \right] \leq \frac{\sqrt{M} (a+T)^{1/6}}{\sqrt{T}} \leq \mathcal{O} \left( \frac{a^{1/6} \sqrt{M}}{\sqrt{T}} + \frac{1}{T^{1/3}} \right) = \mathcal{O} \left( \frac{1}{T^{1/3}} \right),$$

where the last inequality is due to  $(a+b)^{1/3} \leq a^{1/3} + b^{1/3}$ . So, we can achieve the stationary point with  $T = \mathcal{O}(1/\epsilon^3)$ .

## B. Proof of Lemma 1

We first calculate the cumulative variance. Denote that  $\Upsilon_t^i := \|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\|^2$  and  $\Phi_t^i := \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2$ . According to Lemma 6, we have:

$$\sum_{i=1}^K \mathbb{E} [\Upsilon_{t+1}^i + \Phi_{t+1}^i] \leq (1 - \beta) \sum_{i=1}^K \mathbb{E} [\Upsilon_t^i + \Phi_t^i] + 2\beta^2 K (\sigma_J^2 + \sigma_f^2) + 2(\mathcal{L}_J^2 + \mathcal{L}_f^2) \sum_{i=1}^K \mathbb{E} [\|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\|^2]$$

Then, applying Lemma 7 and setting  $\beta \leq \frac{1}{2L_1}$ , we get:

$$\begin{aligned} \sum_{i=1}^K \mathbb{E} [\Upsilon_{t+1}^i + \Phi_{t+1}^i] &\leq (1 - \beta) \sum_{i=1}^K \mathbb{E} [\Upsilon_t^i + \Phi_t^i] + \beta^2 L_1 + L_1 \left( \mathbb{E} [\eta^2 \|\mathbf{v}_t\|^2] + \beta^2 + \beta^2 \sum_{i=1}^K \mathbb{E} [\Upsilon_t^i] \right) \\ &\leq (1 - \frac{\beta}{2}) \sum_{i=1}^K \mathbb{E} [\Upsilon_t^i + \Phi_t^i] + 2\beta^2 L_1 + L_1 \mathbb{E} [\eta^2 \|\mathbf{v}_t\|^2] \end{aligned}$$

By summing up and rearranging, we have:

$$\frac{\beta}{2} \sum_{t=1}^T \sum_{i=1}^K \mathbb{E} [\Upsilon_t^i + \Phi_t^i] \leq \sum_{i=1}^K \mathbb{E} [\Upsilon_1^i + \Phi_1^i] + 2\beta^2 L_1 T + L_1 \eta^2 \sum_{t=1}^T \mathbb{E} [\|\mathbf{v}_t\|^2]$$

Denote that  $\Gamma_t = F(\mathbf{w}_t) + \frac{1}{c_0 \eta} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i)$ . We then try to bound the term  $\sum_{t=1}^T \mathbb{E} [\|\mathbf{v}_t\|^2]$ , which is very similar to the proof of Theorem 1:

$$\begin{aligned} &\sum_{t=1}^T \mathbb{E} [\|\mathbf{v}_t\|^2] \\ &\leq 2 \sum_{t=1}^T \mathbb{E} [\|\nabla F(\mathbf{w}_t)\|^2] + 2 \sum_{t=1}^T \mathbb{E} [\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2] \\ &= \frac{4}{\eta} \left( \frac{\eta}{2} \sum_{t=1}^T \mathbb{E} [\|\nabla F(\mathbf{w}_t)\|^2] + \frac{\eta}{2} \sum_{t=1}^T \mathbb{E} [\|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2] \right) \\ &\leq \mathbb{E} \left[ \frac{4}{\eta} (\Gamma_1 - \Gamma_{T+1}) + 4 \sum_{t=1}^T \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \sum_{t=1}^T \|\mathbf{v}_t\|^2 + \frac{4}{\eta} \sum_{t=1}^T \sum_{i=1}^K \left( \frac{1}{c_0 \eta} (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta} (\Phi_t^i + \Upsilon_t^i) \right) \right] \\ &\leq \mathbb{E} \left[ \frac{4}{\eta} (\Gamma_1 - \Gamma_{T+1}) + \frac{8L_1}{\eta} \sum_{t=1}^T \sum_{i=1}^K \eta (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \|\mathbf{v}_t\|^2 - \sum_{t=1}^T \sum_{i=1}^K \frac{4}{c_0 \eta^2} (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. + \sum_{t=1}^T \sum_{i=1}^K \frac{4(1-\beta)}{c_0 \eta^2} (\Phi_t^i + \Upsilon_t^i) + \sum_{t=1}^T \frac{8\beta^2 K (\sigma_J^2 + \sigma_f^2)}{c_0 \eta^2} + \sum_{t=1}^T \sum_{i=1}^K \frac{8(\mathcal{L}_J^2 + \mathcal{L}_f^2)}{c_0 \eta^2} \|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\|^2 \right] \\ &\leq \mathbb{E} \left[ \frac{4}{\eta} (\Gamma_1 - \Gamma_{T+1}) + \frac{8L_1}{\eta} \sum_{t=1}^T \sum_{i=1}^K \eta (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \|\mathbf{v}_t\|^2 - \sum_{t=1}^T \sum_{i=1}^K \frac{4}{c_0 \eta^2} (\Phi_t^i + \Upsilon_t^i) \right. \\ &\quad \left. + \sum_{t=1}^T \sum_{i=1}^K \frac{4(1-\beta)}{c_0 \eta^2} (\Phi_t^i + \Upsilon_t^i) + \sum_{t=1}^T \frac{8\beta^2 K (\sigma_J^2 + \sigma_f^2)}{c_0 \eta^2} + \sum_{t=1}^T \frac{4L_1}{c_0} \|\mathbf{v}_t\|^2 + \sum_{t=1}^T \frac{4L_1 \beta^2}{c_0 \eta^2} + \sum_{t=1}^T \sum_{i=1}^K \frac{L_1 \beta^2}{c_0 \eta} \Upsilon_t^i \right] \\ &\leq \mathbb{E} \left[ \frac{4}{\eta} (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \sum_{i=1}^K \left( 8L_1 - \frac{2\beta}{c_0 \eta^2} \right) (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \left( 1 - \frac{4L_1}{c_0} \right) \|\mathbf{v}_t\|^2 + \frac{8L_1 \beta^2 T}{c_0 \eta^2} \right] \end{aligned}$$

By setting  $c_0 = 4L_1$  and  $c = 4L_1 c_0 = 16L_1^2$ , where  $L_1$  is defined in the proof of Theorem 1, we get:

$$\sum_{t=1}^T \mathbb{E} [\|\mathbf{v}_t\|^2] \leq \mathbb{E} \left[ \frac{4}{\eta} (\Gamma_1 - \Gamma_{T+1}) + \frac{8L_1 \beta^2 T}{c_0 \eta^2} \right] \leq \mathbb{E} \left[ \frac{4}{\eta} (F(\mathbf{w}_1) - F_*) + \frac{4 \sum_{i=1}^K (\Phi_1^i + \Upsilon_1^i)}{c_0 \eta^2} + \frac{8L_1 \beta^2 T}{c_0 \eta^2} \right]$$

When  $\tau$  is sampled randomly from  $\{1, \dots, T\}$ , we have:

$$\begin{aligned}
 & \sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_\tau^{i-1}) - \mathbf{u}_\tau^i\|^2 \right] + \sum_{i=1}^K \mathbb{E} \left[ \|\mathbf{v}_\tau^i - \nabla f_i(\mathbf{u}_\tau^{i-1})\|^2 \right] \\
 &= \frac{1}{T} \left( \sum_{i=1}^K \sum_{t=1}^T \mathbb{E} \left[ \|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\|^2 \right] + \sum_{i=1}^K \sum_{t=1}^T \mathbb{E} \left[ \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2 \right] \right) \\
 &\leq \mathbb{E} \left[ \frac{2}{\beta T} \sum_{i=1}^K (\Phi_1^i + \Upsilon_1^i) + 4\beta L_1 + \frac{8L_1\eta}{\beta T} (F(\mathbf{w}_1) - F_*) + \frac{8L_1 \sum_{i=1}^K (\Phi_1^i + \Upsilon_1^i)}{c_0\beta T} + \frac{16L_1^2\beta}{c_0} \right] \\
 &\leq \mathbb{E} \left[ \frac{4}{\beta T} \sum_{i=1}^K (\Phi_1^i + \Upsilon_1^i) + 8\beta L_1 + \frac{8L_1\eta}{\beta T} (F(\mathbf{w}_1) - F_*) \right]
 \end{aligned}$$

Let's consider the first stage, in which  $\sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_1^{i-1}) - \mathbf{u}_1^i\|^2 + \|\mathbf{v}_1^i - \nabla f_i(\mathbf{u}_1^{i-1})\|^2 \right] \leq K(\sigma_f^2 + \sigma_j^2)$ . Note that in below the numerical subscripts denote the stage index  $\{1, \dots, S\}$ . With the parameter  $\beta_1 = \frac{1}{2L_1}$ , and  $T_1 = \max \left\{ 2\sqrt{2}L_1\Delta_F, 4L_1K(\sigma_f^2 + \sigma_j^2) \right\}$ , we have:

$$\begin{aligned}
 & \sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_1^{i-1}) - \mathbf{u}_1^i\|^2 \right] + \sum_{i=1}^K \mathbb{E} \left[ \|\mathbf{v}_1^i - \nabla f_i(\mathbf{u}_1^{i-1})\|^2 \right] \\
 &\leq \frac{4}{\beta_1 T_1} K(\sigma_f^2 + \sigma_j^2) + 8\beta_1 L_1 + \frac{8L_1\eta\Delta_F}{\beta_1 T_1} \\
 &\leq 8L_1 \\
 &= \mu\epsilon_1
 \end{aligned}$$

Also, according to the PL condition, we have:

$$\begin{aligned}
 \mathbb{E}[F(\mathbf{w}_1) - F_*] &\leq \mathbb{E} \left[ \frac{\|\nabla F(\mathbf{w}_1)\|^2}{2\mu} \right] \\
 &\leq \frac{4L_1\Delta_F}{\mu T_1\sqrt{\beta_1}} + \frac{4L_1K(\sigma_f^2 + \sigma_j^2)}{\mu\beta_1 T_1} + \frac{8\beta_1 L^2}{\mu} \\
 &\leq \frac{8L_1}{\mu} \\
 &= \epsilon_1
 \end{aligned}$$

Starting from the second stage, we would prove by induction. Suppose at stage  $s-1$ , we have  $F(\mathbf{w}_{s-1}) - F_* \leq \epsilon_{s-1}$ , and  $\sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_{s-1}^{i-1}) - \mathbf{u}_{s-1}^i\|^2 \right] + \sum_{i=1}^K \mathbb{E} \left[ \|\mathbf{v}_{s-1}^i - \nabla f_i(\mathbf{u}_{s-1}^{i-1})\|^2 \right] \leq \mu\epsilon_{s-1}$ . Then at  $s$  stage, with  $\beta_s = \frac{\mu\epsilon_{s-1}}{64L_1^2}$  and  $T_s = \max \left\{ \frac{2048L_1^3}{\mu\epsilon_{s-1}}, \frac{256L_1^2}{u^{3/2}\sqrt{\epsilon_{s-1}}} \right\}$ , we have:

$$\begin{aligned}
 & \sum_{i=1}^K \mathbb{E} \left[ \|f_i(\mathbf{u}_s^{i-1}) - \mathbf{u}_s^i\|^2 \right] + \sum_{i=1}^K \mathbb{E} \left[ \|\mathbf{v}_s^i - \nabla f_i(\mathbf{u}_s^{i-1})\|^2 \right] \\
 &\leq \frac{4\mu\epsilon_{s-1}}{\beta_s T_s} + 8\beta_s L_1 + \frac{8L_1\eta_s\epsilon_{s-1}}{\beta_s T_s} \\
 &\leq \frac{\mu\epsilon_{s-1}}{8} + \frac{\mu\epsilon_{s-1}}{8} + \frac{\mu\epsilon_{s-1}}{8} \\
 &\leq \frac{\mu\epsilon_{s-1}}{2} \\
 &= \mu\epsilon_s
 \end{aligned}$$

The output of stage  $s$  also satisfies the following guarantee:

$$\begin{aligned}
 \mathbb{E}[F(\mathbf{w}_s) - F_*] &\leq \mathbb{E}\left[\frac{\|\nabla F(\mathbf{w}_s)\|^2}{2\mu}\right] \leq \frac{1}{2\mu T_s} \sum_{t=1}^{T_s} \mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] \\
 &\leq \frac{(\Gamma_1 - \Gamma_{T+1})}{\mu\eta_s T_s} + \frac{2L_1\beta_s^2}{\mu c_o \eta_s^2} \\
 &\leq \frac{4L_1(F(\mathbf{w}_{s-1}) - F_*)}{\mu\sqrt{\beta_s} T_s} + \frac{4L_1\mu\epsilon_{s-1}}{\mu\beta_s T_s} + \frac{8L_1^2\beta_s}{\mu} \\
 &\leq \frac{\epsilon_{s-1}}{8} + \frac{\epsilon_{s-1}}{8} + \frac{\epsilon_{s-1}}{8} \\
 &\leq \frac{\epsilon_{s-1}}{2} = \epsilon_s
 \end{aligned}$$

Combining two cases, we have proved that  $\sum_{i=1}^K \mathbb{E}\left[\|f_i(\mathbf{u}_s^{i-1}) - \mathbf{u}_s^i\|^2\right] + \sum_{i=1}^K \mathbb{E}\left[\|\mathbf{v}_s^i - \nabla f_i(\mathbf{u}_s^{i-1})\|^2\right] \leq \mu\epsilon_s$  and  $F(\mathbf{w}_s) - F_* \leq \epsilon_s$ , where  $\epsilon_s = \epsilon_1/2^{s-1}$ . Let  $L_2 = 64L_1^2$ , we have the results in Lemma 1

### C. Proof of Theorem 2

We have proved that  $F(\mathbf{w}_s) - F_* \leq \epsilon_s$  in Lemma 1. That is to say,  $F(\mathbf{w}_S) - F_* \leq \epsilon$  when  $S = \log_2\left(\frac{2\epsilon_1}{\epsilon}\right) = \log_2\left(\frac{L}{\mu\epsilon}\right)$ , and the sample complexity until this stage is computed as:

$$\begin{aligned}
 T_1 + \sum_{s=2}^S T_s &= \mathcal{O}\left(\sum_{s=2}^S T_s\right) \\
 &= \mathcal{O}\left(\sum_{s=2}^S \frac{L_2^{3/2}}{\mu\epsilon_{s-1}} + \frac{L_2}{\mu^{3/2}\sqrt{\epsilon_{s-1}}}\right) \\
 &\leq \mathcal{O}\left(\frac{L_2^{3/2}}{\mu\epsilon}, \frac{L_2}{\mu^{3/2}\sqrt{\epsilon}}\right) \stackrel{\mu \geq \epsilon}{\leq} \mathcal{O}\left(\frac{1}{\mu\epsilon}\right)
 \end{aligned}$$

### D. Proof of Theorem 3

When  $F(\mathbf{w})$  is convex, we define  $\hat{F}(\mathbf{w}) = F(\mathbf{w}) + \frac{\mu}{2}\|\mathbf{w}\|^2$ . We know that  $\hat{F}(\mathbf{w})$  is  $\mu$ -strongly convex, which implies  $\mu$ -PL condition. In Theorem 2, we have proved: for any  $\delta > 0$ , there exist  $T = \mathcal{O}\left(\frac{1}{\mu\delta}\right)$  such that  $\hat{F}(\mathbf{w}_T) - \hat{F}_* \leq \delta$ . It indicates that  $F(\mathbf{w}_T) - F_* \leq \delta + \frac{\mu}{2}\|\mathbf{w}_*\|^2 - \frac{\mu}{2}\|\mathbf{w}_T\|^2 \leq \delta + \frac{\mu}{2}D$ . For any  $\epsilon > 0$ , if we choose  $\mu = \frac{\epsilon}{D}$  and  $\delta = \frac{\epsilon}{2}$ , we get  $F(\mathbf{w}_T) - F_* \leq \epsilon$ , for some  $T = \mathcal{O}\left(\frac{2D}{\epsilon^2}\right) = \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ .

### E. Proof of Theorem 4

Note that since the norm of estimated gradient  $\|\mathbf{v}_t\|$  is bounded, the value of the learning rate scaling factor  $\mathbf{c} = 1/(\sqrt{\mathbf{h}_t} + \delta)$  presented in (5) is also upper bounded and lower bounded, which can be presented as  $c_l \leq \|\mathbf{c}\|_\infty \leq c_u$ . With this property, We can introduce the variant version of Lemma 3, which can also be found in Lemma 3 in (Guo et al., 2021).

**Lemma 8.** For  $\mathbf{w}_{t+1} = \mathbf{w}_t - \tilde{\eta}_t \mathbf{v}_t$ , with  $\eta_t c_l \leq \tilde{\eta}_t \leq \eta_t c_u$  and  $\eta_t L_F \leq c_l/2c_u^2$ , we have following guarantee:

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta_t c_u}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t c_l}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2.$$

*Proof.* Due to the smoothness of function  $F$  and the definition of  $\mathbf{w}_{t+1}$ , we have:

$$\begin{aligned}
 F(\mathbf{w}_{t+1}) &\leq F(\mathbf{w}_t) + \langle \nabla F(\mathbf{w}_t), \mathbf{w}_{t+1} - \mathbf{w}_t \rangle + \frac{L_F}{2} \|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 \\
 &= F(\mathbf{w}_t) - \tilde{\eta}_t \langle \nabla F(\mathbf{w}_t), \mathbf{v}_t \rangle + \frac{\tilde{\eta}_t^2 L_F}{2} \|\mathbf{v}_t\|^2 \\
 &\leq F(\mathbf{w}_t) - \tilde{\eta}_t \langle \nabla F(\mathbf{w}_t), \mathbf{v}_t \rangle + \frac{\tilde{\eta}_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 + \frac{\tilde{\eta}_t}{2} \|\mathbf{v}_t\|^2 - \frac{\tilde{\eta}_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\tilde{\eta}_t}{2} \|\mathbf{v}_t\|^2 + \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2 \\
 &= F(\mathbf{w}_t) + \frac{\tilde{\eta}_t}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\tilde{\eta}_t}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2 \\
 &\leq F(\mathbf{w}_t) + \frac{\eta_t c_u}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t c_l}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2
 \end{aligned}$$

□

We set

$$L_3 = \max \left\{ 1, c_u K L_f^{2(K-1)}, c_u K L_F^2, 2K(\sigma_f^2 + \sigma_j^2), 2 \left( 1 + \frac{1}{c_l} \right) (L_f^2 + L_f^2) (1 + 2K + 2K\sigma_f^2) \sum_{i=1}^K (2\mathcal{L}_f^2)^{i-1} \right\}.$$

Similar to the proof of Theorem 1, we denote that  $\Gamma_t = F(\mathbf{w}_t) + \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i)$ ,  $\Phi_t^i := \|\mathbf{v}_t^i - \nabla f_i(\mathbf{u}_t^{i-1})\|^2$  and  $\Upsilon_t^i := \mathbb{E} \left[ \|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\|^2 \right]$ . Based on Lemma 8, we have:

$$\begin{aligned}
 &\mathbb{E} [\Gamma_{t+1} - \Gamma_t] \\
 &= \mathbb{E} \left[ F(\mathbf{w}_{t+1}) - F(\mathbf{w}_t) + \frac{1}{c_0 \eta_t} \sum_{i=1}^K (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i) \right] \\
 &\leq \mathbb{E} \left[ \frac{\eta_t c_u}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \frac{\eta_t c_l}{2} \|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2 + \frac{1}{c_0 \eta_t} \sum_{i=1}^K (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta_{t-1}} \sum_{i=1}^K (\Phi_t^i + \Upsilon_t^i) \right]
 \end{aligned}$$

By summing up and rearranging, we have:

$$\begin{aligned}
 &\mathbb{E} \left[ \sum_{t=1}^T \frac{\eta_t c_l}{2} \|\nabla F(\mathbf{w}_t)\|^2 \right] \\
 &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \frac{\eta_t c_u}{2} \|\nabla F(\mathbf{w}_t) - \mathbf{v}_t\|^2 - \sum_{t=1}^T \frac{\eta_t c_l}{4} \|\mathbf{v}_t\|^2 \right. \\
 &\quad \left. + \sum_{t=1}^T \sum_{i=1}^K \left( \frac{1}{c_0 \eta_t} (\Phi_{t+1}^i + \Upsilon_{t+1}^i) - \frac{1}{c_0 \eta_{t-1}} (\Phi_t^i + \Upsilon_t^i) \right) \right] \\
 &\leq \mathbb{E} \left[ (\Gamma_1 - \Gamma_{T+1}) + \sum_{t=1}^T \sum_{i=1}^K \left( L_3 \eta_t - \frac{1}{c_0 \eta_{t-1}} + \frac{L_3 \beta_{t+1}^2}{c_0 \eta_t} + \frac{(1 - \beta_{t+1})}{c_0 \eta_t} \right) (\Phi_t^i + \Upsilon_t^i) - \sum_{t=1}^T \left( \frac{c_l}{4} - \frac{L_3 c_l}{c_0} \right) \eta_t \|\mathbf{v}_t\|^2 \right. \\
 &\quad \left. + \sum_{t=1}^T \frac{2L_3 \beta_{t+1}^2}{c_0 \eta_t} \right]
 \end{aligned}$$

Set  $c_0 = 4L_3$ ,  $\eta_t = (a + t)^{-1/3}$  and  $c = 10L_3^2$ ,  $a = (20L_3^3)^{3/2}$ , we have:

$$\mathbb{E} \left[ \sum_{t=1}^T \frac{\eta_t c_l}{2} \|\nabla F(\mathbf{w}_t)\|^2 \right] \leq \Delta_F + \frac{K(\sigma_f^2 + \sigma_j^2)}{c_0 \eta_0} + 50L_3^4 \ln(T + 1)$$

Since  $\eta_t$  is decreasing, we have:

$$\eta_T \sum_{t=1}^T \|\nabla F(\mathbf{w}_t)\|^2 \leq \frac{1}{c_t} \left( 2\Delta_F + \frac{2K(\sigma_f^2 + \sigma_j^2)}{c_0\eta_0} + 100L_3^4 \ln(T+1) \right).$$

Very similar to the proof of Theorem 1, set  $M = \frac{1}{c_t} \left( 2\Delta_F + \frac{2K(\sigma_f^2 + \sigma_j^2)}{c_0\eta_0} + 100L_3^4 \ln(T+1) \right)$  and we have:

$$\mathbb{E} \left[ \sum_{t=1}^T \frac{\|\nabla F(\mathbf{w}_t)\|}{T} \right] \leq \frac{\sqrt{M}(a+T)^{1/6}}{\sqrt{T}} \leq \mathcal{O} \left( \frac{a^{1/6}\sqrt{M}}{\sqrt{T}} + \frac{1}{T^{1/3}} \right) = \mathcal{O} \left( \frac{1}{T^{1/3}} \right).$$

So, Adaptive SMVR enjoys the same  $\mathcal{O}(1/\epsilon^3)$  sample complexity as the SMVR method.