On the Convergence of Local Stochastic Compositional Gradient Descent with Momentum

Hongchang Gao [1] **Junyi Li** [2] **Heng Huang** [2]

Abstract

Federated Learning has been actively studied due to its efficiency in numerous real-world applications in the past few years. However, the federated stochastic compositional optimization problem is still underexplored, even though it has widespread applications in machine learning. In this paper, we developed a novel local stochastic compositional gradient descent with momentum method, which facilitates Federated Learning for the stochastic compositional problem. Importantly, we investigated the convergence rate of our proposed method and proved that it can achieve the $O(1/\epsilon^4)$ sample complexity, which is better than existing methods. Meanwhile, our communication complexity $O(1/\epsilon^3)$ can match existing methods. To the best of our knowledge, this is the first work achieving such favorable sample and communication complexities. Additionally, extensive experimental results demonstrate the superior empirical performance over existing methods, confirming the efficacy of our method.

1. Introduction

Federated Learning has attracted increasing attention in recent years. It facilitates the distributed data analysis without sharing the raw data. Thus, it has been applied to various machine learning tasks. However, most existing works just focus on the standard stochastic minimization problem, ignoring the stochastic compositional optimization problem. In fact, numerous machine learning models can be formulated as the stochastic compositional optimization problem, such as the model-agnostic meta-learning (MAML) problem (Finn et al., 2017), the imbalanced image classification problem (Qi et al., 2021). To bridge this gap, in this paper we consider the federated compositional optimization problem as follows:

$$\min_{x\in\mathbb{R}^d} \frac{1}{K}\sum_{k=1}^{K} \mathbb{E}_{\zeta\sim\mathcal{D}_f^{(k)}}\left[f^{(k)}\left(\mathbb{E}_{\xi\sim\mathcal{D}_g^{(k)}}[g^{(k)}(x;\xi)];\zeta\right)\right]. \tag{1}$$

Here, $f^{(k)}(y) \triangleq \mathbb{E}_{\zeta\sim\mathcal{D}_f^{(k)}}[f^{(k)}(y;\zeta)] \in \mathbb{R}$ is the outer-level function on the k-th device where $y \in \mathbb{R}^{d'}$ and $\mathcal{D}_f^{(k)}$ denotes the data distribution for the outer-level function on the k-th device. $g^{(k)}(x) \triangleq \mathbb{E}_{\xi\sim\mathcal{D}_g^{(k)}}[g^{(k)}(x;\xi)] \in \mathbb{R}^{d'}$ is the inner-level function on the k-th device where $x \in \mathbb{R}^d$ and $\mathcal{D}_g^{(k)}$ is the data distribution for the inner-level function on the k-th device. It can be observed that there are two stochastic functions in this optimization problem, which is different from the standard Federated Learning model.

Due to the widespread application of the stochastic compositional optimization problem in machine learning, a lot of efforts (Wang et al., 2017a;b; Zhang & Xiao, 2019a; Yuan et al., 2019; Yang & Hu, 2020) have been made to develop efficient optimization algorithms for solving Eq. (1) when $K = 1$. Since there are two level stochastic functions in Eq. (1), the standard stochastic gradient is a biased estimation of the full gradient when the outer-level function is nonlinear. As a result, stochastic gradient descent (SGD) converges slowly when optimizing Eq. (1). To address this issue, (Wang et al., 2017a) developed the stochastic compositional gradient descent (SCGD) method by introducing the moving average estimation for the inner-level function to improve the convergence performance, whose sample complexity to achieve the ϵ-accuracy solution is $O(1/\epsilon^8)$ for nonconvex problems. Afterwards, a series of works, such as (Zhang & Xiao, 2019a;b; Yuan et al., 2019; Yuan & Hu, 2020; Qi et al., 2020), focus on further improving the sample complexity of SCGD by incorporating the acceleration or variance reduction techniques. However, the single-machine setting is different from Federated Learning so that it is unclear how to apply these methods to Federated Learning and how they converge. Especially, it is unclear how the communication period in Federated Learning affects their convergence rates.

On the other hand, numerous federated optimization meth-

[1]Department of Computer and Information Sciences, Temple University, PA, USA. [2]Department of Electrical and Computer Engineering, University of Pittsburgh, PA, USA. Correspondence to: Hongchang Gao <hongchang.gao@temple.edu>, Junyi Li <junyili.ai@gmail.com>, Heng Huang <heng.huang@pitt.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

ods have been explored in recent years. The essential idea of federated optimization methods is to conduct multiple local updates and then perform communication. Under this setting, the computation/sample and communication complexities of federated optimization methods have been extensively studied. For instance, (Stich, 2018) studied the convergence rate of local SGD for strongly convex problems. (Yu et al., 2019b;a) established the convergence rate of local SGD and momentum local SGD for nonconvex problems. Moreover, a series of methods have been proposed to address the heterogeneous data distribution issue (Karimireddy et al., 2020b; Murata & Suzuki, 2021; Li et al., 2020) and alleviate the communication issue (Basu et al., 2019; Rothchild et al., 2020; Reisizadeh et al., 2020; Gao et al., 2021). However, these works restrict their focus on the standard minimization problem so that it is not appropriate to utilize them to solve Eq. (1). Recently, (Huang et al., 2021) developed a local biased SGD method for optimizing Eq. (1). This method just employed standard stochastic gradient so that the bias caused by the compositional structure in the loss function degenerates the convergence rate. Specifically, its sample complexity is $O(1/\epsilon^8)$ and communication complexity is $O(1/\epsilon^4)$ for nonconvex problems. Another recent work (Wang et al., 2021) formulated the model personalization in Federated Learning as a stochastic compositional optimization, and proposed a local SCGD method whose sample complexity is $O(1/\epsilon^5)$ and communication complexity is $O(1/\epsilon^3)$. Obviously, these two methods' sample complexity is much worse than $O(1/\epsilon^4)$ of existing single-machine SCGD methods (Ghadimi et al., 2020; Chen et al., 2020; 2021b). Therefore, a natural question follows: *Is it possible to have a local SCGD method which can achieve a better sample complexity than existing federated compositional optimization methods?*

In this paper, we provide an affirmative answer for the aforementioned question. In particular, we developed a novel local stochastic compositional gradient descent method with momentum (Local-SCGDM) for optimizing Eq. (1). In particular, Local-SCGDM demonstrates how to apply the momentum technique to SCGD for federated compositional optimization problems, such as what variables should be communicated. Importantly, the convergence rate of Local-SCGDM is improved significantly compared with existing methods. In detail, our theoretical results show that, by setting the batch size to $O(1)$, our method can achieve $O(1/\sqrt{T})$ convergence rate where T is the number of iterations. Consequently, our Local-SCGDM can achieve the $O(1/\epsilon^4)$ sample complexity, which is much better than existing methods (Wang et al., 2021; Huang et al., 2021). Additionally, the communication period of our method can be as large as $O(T^{1/4})$, resulting in a $O(1/\epsilon^3)$ communication complexity, which can match the existing method (Wang et al., 2021). To the best of our knowledge, this is the first work achieving such nice sample and communication complexities for the federated compositional optimization problem. At last, we conduct extensive experiments on federated model-agnostic meta-learning task, whose results confirm the efficacy of our proposed method. In summary, our work made the following contributions.

- We developed a novel Local-SCGDM method for optimizing the federated compositional optimization problem. This is the first time to show how the momentum technique is used in this setting.
- Our Local-SCGDM can achieve the $O(1/\epsilon^4)$ sample complexity and $O(1/\epsilon^3)$ communication complexity with $O(1)$ batch size, which improves existing complexities significantly.
- We conduct extensive experiments on federated model-agnostic meta-learning problems. The experimental results validate the superiority of our theoretical results.

2. Related Work

2.1. Stochastic Compositional Optimization Methods

Since there are two levels of stochastic functions in Eq. (1), the standard stochastic gradient is a biased estimation for the full gradient. As a result, directly using SGD to optimize Eq. (1) converges slowly. To address this problem, (Wang et al., 2017a) developed SCGD. In particular, SCGD utilizes the moving average technique to estimate the inner-level function value, based on which the stochastic gradient of the outer-level function is computed. Specifically, if we consider the single-machine setting, SCGD updates the model parameter as follows:

$$\begin{aligned} u_{t+1} &= (1-\gamma)u_t + \gamma g(x_t;\xi)\ , \\ z_t &= \nabla g(x_t;\xi)^T \nabla_g f(u_t;\zeta) \\ x_{t+1} &= x_t - \eta z_t\ , \end{aligned} \tag{2}$$

where $\gamma \in (0,1)$ and $\eta > 0$. Here, u_t is the moving average of the inner-level function value. Then, the stochastic gradient of the outer-level function is evaluated on u_t rather than $g(x_t;\xi)$. In this way, it can reduce the estimation variance. (Wang et al., 2017a) proved that the sample complexity of SCGD is $O(1/\epsilon^8)$ when optimizing the nonconvex problem. Afterwards, (Wang et al., 2017b) developed an accelerated SCGD method, which applies the extrapolation-smoothing scheme to x_t for accelerating the convergence speed. As a result, it can achieve the $O(1/\epsilon^{4.5})$ sample complexity for nonconvex problems.

Inspired by the variance reduction technique in the non-compositional stochastic optimization field, a couple of works (Zhang & Xiao, 2019b; Yuan et al., 2019) propose to accelerate SCGD by reducing the gradient variance. For

instance, (Zhang & Xiao, 2019b) developed a composite incremental variance-reduced (CIVR) method by employing the SPIDER variance reduction technique (Fang et al., 2018; Nguyen et al., 2017), which can achieve the $O(1/\epsilon^3)$ sample complexity for nonconvex problems. However, these methods need to periodically compute the full gradient so that they are not applicable to large-scale applications. Recently, (Chen et al., 2020) developed SCSC by employing the STORM variance reduction technique (Cutkosky & Orabona, 2019) to estimate the inner-level function value and achieved the $O(1/\epsilon^4)$ sample complexity. (Ghadimi et al., 2020) applied the momentum technique to SCGD, which also achieved the $O(1/\epsilon^4)$ sample complexity. More recently, (Chen et al., 2021a) studied the convergence rate of SCGD from the perspective of nested optimization and improved the sample complexity of traditional SCGD (Wang et al., 2017a) to $O(1/\epsilon^4)$.

2.2. Federated Optimization Methods

With the development of Federated Learning, many local SGD methods have been developed in recent years. For instance, (Stich, 2018) established the sample complexity and communication complexity of local SGD for strongly-convex problems. (Yu et al., 2019b) provided the sample complexity and communication complexity for nonconvex problems whose stochastic gradients have bounded second moment. Later, (Yu et al., 2019a) remove the bounded second moment assumption and established the convergence rate of momentum local SGD for nonconvex problems. In addition, (Xu et al., 2021; Liu et al., 2020; Gao & Huang, 2020) also studied the momentum technique for local SGD under different settings and established the convergence rate. Furthermore, much progress has been made to improve the communication complexity of local SGD by compressing gradients (Basu et al., 2019; Rothchild et al., 2020; Reisizadeh et al., 2020; Gao et al., 2021) and reducing the variance of stochastic gradients (Khanduri et al., 2021; Karimireddy et al., 2020a; Das et al., 2020). However, all these methods restrict their focus on the standard minimization problem. Therefore, designing efficient local SCGD methods is necessary and important. Recently, (Huang et al., 2021) studied the stochastic compositional problem for Federated Learning. Specifically, γ in Eq. (2) is set to 1 in (Huang et al., 2021). As such, it is a standard stochastic gradient descent method and the large bias of $\nabla g(x_t;\xi)^T\nabla_g f(g(x_t;\xi);\zeta)$ slows down the convergence rate, resulting in the $O(1/\epsilon^8)$ sample complexity and $O(1/\epsilon^4)$ communication complexity for nonconvex problems. As for the recent work (Wang et al., 2021), they viewed the model personalization problem in Federated Learning as a model-agnostic meta-learning problem, and then utilized SCGD to solve this problem, rather than SGD as (Huang et al., 2021). As such, the sample and communication complexities are improved to $O(1/\epsilon^5)$ and $O(1/\epsilon^3)$, respectively. However, this method has a limitation. It requires to *maintain an inner state u_t for each task*. As a result, it is not applicable to large-scale settings due to the large memory complexity.

3. Local Stochastic Compositional Gradient Descent with Momentum

In this section, we present the algorithmic details of local stochastic compositional gradient descent with momentum, its convergence rate, as well as its application to federated model-agnostic meta-learning.

Algorithm 1 Local-SCGDM

Input: $\eta>0, \beta>0, \gamma>0, \alpha>0, p>1, x_0^{(k)}=x_0$

1: **for** $t=0,\cdots,T-1$ **do**
2: **if** $t==0$ **then**
3: $u_1^{(k)}=g^{(k)}(x_0^{(k)};\xi_0^{(k)})$,
4: $z_0^{(k)}=\nabla g^{(k)}(x_0^{(k)};\xi_0^{(k)})^T\nabla_g f^{(k)}(u_1^{(k)};\zeta_0^{(k)})$,
5: $m_1^{(k)}=z_0^{(k)}$,
6: **else**
7: $u_{t+1}^{(k)}=(1-\gamma\eta)u_t^{(k)}+\gamma\eta g^{(k)}(x_t^{(k)};\xi_t^{(k)})$,
8: $z_t^{(k)}=\nabla g^{(k)}(x_t^{(k)};\xi_t^{(k)})^T\nabla_g f^{(k)}(u_{t+1}^{(k)};\zeta_t^{(k)})$,
9: $m_{t+1}^{(k)}=(1-\alpha\eta)m_t^{(k)}+\alpha\eta z_t^{(k)}$,
10: **end if**
11: $x_{t+1}^{(k)}=x_t^{(k)}-\beta\eta m_{t+1}^{(k)}$,
12: **if** $\mathrm{mod}(t+1,p)==0$ **then**
13: $u_{t+1}^{(k)}=\bar{u}_{t+1}\triangleq\frac{1}{K}\sum_{k'=1}^K u_{t+1}^{(k')}$,
14: $m_{t+1}^{(k)}=\bar{m}_{t+1}\triangleq\frac{1}{K}\sum_{k'=1}^K m_{t+1}^{(k')}$,
15: $x_{t+1}^{(k)}=\bar{x}_{t+1}\triangleq\frac{1}{K}\sum_{k'=1}^K x_{t+1}^{(k')}$,
16: **end if**
17: **end for**

3.1. Local Stochastic Compositional Gradient Descent with Momentum

The key idea of our proposed local stochastic compositional gradient descent with momentum (Local-SCGDM) method is to utilize momentum SCGD to update model parameters on each device for multiple iterations, and then the communication is conducted between devices and the central server. The detail of Local-SCGDM is presented in Algorithm 1. Specifically, in the first iteration, each device utilizes the standard stochastic gradient descent method to update the model parameter, which can be found in Lines 3-5 of Algorithm 1. In other iterations, each worker computes the momentum $m_{t+1}^{(k)}$ for the stochastic compositional gradient $\nabla g^{(k)}(x_t^{(k)};\xi_t^{(k)})^T\nabla_g f^{(k)}(u_{t+1}^{(k)};\zeta_t^{(k)})$ to update the model parameter $x_{t+1}^{(k)}$, which can be found in Lines 7-9 of Algorithm 1. Here, since the moving average strategy is used

for the update of $u_{t+1}^{(k)}$ and $m_{t+1}^{(k)}$, their coefficients should satisfy $0 < \gamma\eta < 1$ and $0 < \alpha\eta < 1$. Following the scheme of Federated Learning, at every p iterations, all devices perform communication with the central server. Here, the communication period p is greater than 1 such that the number of communication rounds is reduced to T/p. It is worth noting that Local-SCGDM communicates the model parameter $x_{t+1}^{(k)}$, momentum $m_{t+1}^{(k)}$, and inner-level function estimator $u_{t+1}^{(k)}$ with the central server, which is inspired by our theoretical analysis. Additionally, the standard momentum local SGD (Yu et al., 2019a) also requires to communicate momentum. Thus, the communication strategy in our method is reasonable.

To sum up, we developed a novel Local-SCGDM method for solving the federated compositional optimization problem in Eq. (1). To the best of our knowledge, this is the first work applying the momentum technique to the federated compositional optimization problem for improving its efficiency. Importantly, our method discloses what variables should be communicated, which has been ignored in existing literature.

3.2. Convergence Rate

Before presenting the convergence rate of our proposed Local-SCGDM, we first introduce the following assumptions that are widely used in existing stochastic compositional optimization methods.

Assumption 3.1. (Smoothness) For any $k \in \{1, 2, \cdots, K\}$, the function $g^{(k)}(\cdot)$ is L_g-Lipschitz smooth and the function $f^{(k)}(\cdot)$ is L_f-Lipschitz smooth, i.e., for any $x_1, x_2 \in \operatorname{dom} g^{(k)}$, and any $y_1, y_2 \in \operatorname{dom} f^{(k)}$, there exist $L_g > 0$ and $L_f > 0$ such that

$$\begin{aligned} \|\nabla g^{(k)}(x_1) - \nabla g^{(k)}(x_2)\| &\leq L_g \|x_1 - x_2\|, \\ \|\nabla f^{(k)}(y_1) - \nabla f^{(k)}(y_2)\| &\leq L_f \|y_1 - y_2\|. \end{aligned} \tag{3}$$

Assumption 3.2. (Bounded gradient) For any $k \in \{1, 2, \cdots, K\}$, the function $g^{(k)}(\cdot)$ is C_g-Lipschitz continuous and the function $f^{(k)}(\cdot)$ is C_f-Lipschitz continuous where $C_g > 0$ and $C_f > 0$. Additionally, for any $x \in \operatorname{dom} g^{(k)}$ and $y \in \operatorname{dom} f^{(k)}$, the second moments of $\nabla f^{(k)}(y; \zeta)$ and $\nabla g^{(k)}(x; \xi)$ are bounded as follows:

$$\begin{aligned} \mathbb{E}_\xi[\|\nabla g^{(k)}(x; \xi)\|^2] &\leq C_g^2, \\ \mathbb{E}_\zeta[\|\nabla f^{(k)}(y; \zeta)\|^2] &\leq C_f^2 \,. \end{aligned} \tag{4}$$

Assumption 3.3. (Bounded variance) For any $k \in \{1, 2, \cdots, K\}$, $x \in \operatorname{dom} g^{(k)}$, and $y \in \operatorname{dom} f^{(k)}$, there exist constant values $\sigma_f > 0$, $\sigma_g > 0$, $\delta_g > 0$ such that

$$\begin{aligned} &\mathbb{E}_\zeta[\|\nabla f^{(k)}(y; \zeta) - \nabla f^{(k)}(y)\|^2] \leq \sigma_f^2, \\ &\mathbb{E}_\xi[\|\nabla g^{(k)}(x; \xi) - \nabla g^{(k)}(x)\|^2] \leq \sigma_g^2, \\ &\mathbb{E}_\xi[\|g^{(k)}(x; \xi) - g^{(k)}(x)\|^2] \leq \delta_g^2. \end{aligned} \tag{5}$$

Additionally, following existing stochastic compositional optimization methods, we denote $F^{(k)}(x) \triangleq f^{(k)}(g^{(k)}(x))$ and $F(x) = \frac{1}{K}\sum_{k=1}^K F^{(k)}(x)$ which is L_F-smooth with $L_F = C_g^2 L_f + C_f L_g$. Then, we provide the convergence rate for Algorithm 1 in the following theorem.

Theorem 3.4. *Suppose Assumption 3.1-3.3 hold, if* $\alpha < \frac{1}{\eta}$, $\gamma < \frac{1}{\eta}$, $\beta \leq \sqrt{\frac{1}{8}\Big/\Big(\frac{C_g^4 L_f^2}{\gamma^2} + \frac{C_g^4 L_f^2}{\gamma} + \frac{L_F^2}{\alpha^2}\Big)}$, $\eta \leq \min\{1, \frac{1}{2\beta(C_g^2 L_f + C_f L_g)}\}$, $p \leq \frac{1}{4\beta\eta\sqrt{6C_f^2 L_g^2 + 16 C_g^4 L_f^2}}$, *Algorithm 1 has the following convergence rate:*

$$\begin{aligned} \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] \leq{}& \frac{2(F(x_0) - F(x_*))}{\beta\eta T} \\ &+ \frac{6(C_g^2\sigma_f^2 + C_g^2 L_f^2\delta^2 + C_f^2\sigma_g^2)}{\alpha\eta T} + \frac{4C_g^2 L_g^2\delta^2}{\gamma\eta T} \\ &+ 2\gamma\eta C_g^2 L_f^2\delta_g^2 + 2\alpha\eta(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\gamma^2\eta^2 C_g^2 L_f^2\delta_g^2 \\ &+ 32p^2\beta^2\eta^2 L_F^2(16C_g^2 L_f^2\delta_g^2 + C_g^2\sigma_f^2 + 3C_f^2\sigma_g^2)\,. \end{aligned} \tag{6}$$

Corollary 3.5. *Suppose Assumption 3.1-3.3 hold, by setting* $\eta = T^{-1/2}$, $p = T^{1/4}$, *Algorithm 1 has the following convergence rate:*

$$\begin{aligned} \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] \leq{}& \frac{2(F(x_0) - F(x_*))}{\beta\sqrt{T}} \\ &+ \frac{6(C_g^2\sigma_f^2 + C_g^2 L_f^2\delta^2 + C_f^2\sigma_g^2)}{\alpha\sqrt{T}} + \frac{4C_g^2 L_g^2\delta^2}{\gamma\sqrt{T}} \\ &+ \frac{2\gamma C_g^2 L_f^2\delta_g^2}{\sqrt{T}} + \frac{2\alpha(C_g^2\sigma_f^2 + C_f^2\sigma_g^2)}{\sqrt{T}} + \frac{2\gamma^2 C_g^2 L_f^2\delta_g^2}{T} \\ &+ \frac{32\beta^2 L_F^2(16C_g^2 L_f^2\delta_g^2 + C_g^2\sigma_f^2 + 3C_f^2\sigma_g^2)}{\sqrt{T}}. \end{aligned} \tag{7}$$

Remark 3.6. The hyperparameters α, γ, β in Corollary 3.5 are some constant values. They do not affect the order of the convergence rate. For instance, when setting $\alpha = 1$ and $\gamma = 1$, we have $\beta = \frac{1}{\sqrt{16C_g^4 L_f^2 + 8L_F^2}}$. Algorithm 1 still enjoys the $O(\frac{1}{\sqrt{T}})$ convergence rate.

Remark 3.7. According to Corollary 3.5, to make $\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla F(\bar{x}_t)\| \leq \epsilon$, T should be as large as $O(1/\epsilon^4)$. Because the batch size of our method is just $O(1)$, the sample complexity is $O(1/\epsilon^4)$, which is much better than $O(1/\epsilon^8)$ (Huang et al., 2021) and $O(1/\epsilon^5)$ (Wang et al., 2021). Additionally, the communication complexity of our method is $T/p = O(1/\epsilon^3)$, which can match that in (Wang et al., 2021) and better than $O(1/\epsilon^4)$ in (Huang et al., 2021).

3.3. Application: Federated Model-Agnostic Meta-Learning

In this subsection, we apply our Local-SGDM method to the federated model-agnostic meta-learning problem.

Model-agnostic meta-learning (MAML) (Finn et al., 2017) aims to learn a meta-initialization model that can be easily adapted to new tasks. In Federated MAML, it is assumed that each device has a set of tasks and the goal is to learn a common meta-initialization model by the collaboration between all devices. Formally, the loss function on each device is defined as follows:

$$\begin{aligned}
&\min_{x\in\mathbb{R}^d} \frac{1}{K}\sum_{k=1}^{K} F^{(k)}(x) \triangleq \frac{1}{K}\sum_{k=1}^{K} f^{(k)}(g^{(k)}(x)) \,, \\
&\text{where } g^{(k)}(x) = \mathbb{E}_{\xi^{(k)}\sim\mathcal{D}^{(k)}_{i,\text{train}}}\left[g^{(k)}(x;\xi^{(k)})\right] \\
&\qquad = \mathbb{E}_{\xi^{(k)}\sim\mathcal{D}^{(k)}_{i,\text{train}}}\left[x - \lambda\nabla\mathcal{L}^{(k)}_i\left(x;\xi^{(k)}\right)\right] \,, \\
&f^{(k)}(x) = \mathbb{E}_{i\sim\mathcal{P}^{(k)}_{\text{task}},\zeta^{(k)}\sim\mathcal{D}^{(k)}_{i,\text{test}}}\left[f^{(k)}(y;\zeta^{(k)})\right] \\
&\quad = \mathbb{E}_{i\sim\mathcal{P}^{(k)}_{\text{task}},\zeta^{(k)}\sim\mathcal{D}^{(k)}_{i,\text{test}}}\left[\mathcal{L}^{(k)}_i\left(y;\zeta^{(k)}\right)\right] \,.
\end{aligned} \tag{8}$$

Here, $\mathcal{L}^{(k)}_i$ denotes the loss function for the i-th task on the k-th device, $\lambda > 0$ denotes the step size, $\mathcal{P}^{(k)}_{\text{task}}$ represents the task distribution on the k-th device, $\mathcal{D}^{(k)}_{i,\text{train}}$ and $\mathcal{D}^{(k)}_{i,\text{test}}$ represent the training and test set of the i-th task on the k-th device, respectively. From Eq. (8), it can be observed that Federated MAML is an federated compositional optimization problem. Thus, it can be optimized by our Algorithm 1.

In the following, we demonstrate that Theorem 1 holds for Federated MAML. Here, we assume the loss function $\mathcal{L}^{(k)}_i$ satisfies the following assumptions, which have been widely used in existing literature (Wang et al., 2021; Ji et al., 2020).

Assumption 3.8. For any $k \in \{1, 2, \cdots, K\}$, $(x, y) \in \text{dom}\, \mathcal{L}^{(k)}_i$, we have

$$\begin{aligned}
&\|\nabla\mathcal{L}^{(k)}_i(x) - \nabla\mathcal{L}^{(k)}_i(y)\| \le L_1\|x-y\| \,, \\
&\|\nabla^2\mathcal{L}^{(k)}_i(x) - \nabla^2\mathcal{L}^{(k)}_i(y)\| \le L_2\|x-y\| \,, \\
&\mathbb{E}[\|\nabla\mathcal{L}^{(k)}_i(x;\xi) - \nabla\mathcal{L}^{(k)}_i(x)\|^2] \le \sigma_1^2 \,, \\
&\mathbb{E}[\|\nabla^2\mathcal{L}^{(k)}_i(x;\xi) - \nabla^2\mathcal{L}^{(k)}_i(x)\|^2] \le \sigma_2^2 \,, \\
&\|\nabla\mathcal{L}^{(k)}_i(x)\| \le G \,,
\end{aligned} \tag{9}$$

where L_1, L_2, σ_1, σ_2, G are all positive constants.

Based on Assumption 3.8, we can get the following lemma for the properties of function $f^{(k)}$ and $g^{(k)}$ in Eq. (8).

Lemma 3.9. *Given Assumption 3.8, we can get*

$$\begin{aligned}
&L_f = L_1, L_g = \lambda L_2, \\
&C_f^2 = \sigma_1^2 + G^2, C_g^2 = (1+\lambda L_1)^2 + \lambda^2\sigma_2^2, \\
&\sigma_f^2 = \sigma_1^2, \sigma_g^2 = \lambda^2\sigma_2^2, \delta_g^2 = \lambda^2\sigma_1^2 \,, \\
&\|\nabla F^{(k)}(x) - \nabla F^{(k)}(y)\| \\
&\le ((1+\lambda L_1)^2 L_1 + \lambda G L_2)\|x-y\| \,.
\end{aligned} \tag{10}$$

From Lemma 3.9, we can conclude that all assumptions in Theorem 3.4 are satisfied for Federated MAML in Eq. (8). Thus, the sample complexity and communication complexity of our Local-SCGDM for Federated MAML are $O(1/\epsilon^4)$ and $O(1/\epsilon^3)$, respectively.

4. Proof Sketch

In this section, we present the proof sketch of Theorem 3.4. The details are deferred to Supplementary Materials.

The key idea of our proof is to bound the difference between momentum and gradients $\mathbb{E}[\|m^{(k)}_{t+1} - \nabla F^{(k)}(x^{(k)}_t)\|^2]$, the variance of inner function estimator $\mathbb{E}[\|u^{(k)}_{t+1} - g^{(k)}(x^{(k)}_t)\|^2]$, as well as the difference between the local model and the averaged model $\mathbb{E}[\|\bar{x}_t - x^{(k)}_t\|^2]$. Here, we construct a novel potential function to complete the proof.

It is worth noting that our proof is different from the standard local stochastic gradient descent with momentum method (Yu et al., 2019a). Specifically, (Yu et al., 2019a) introduced a virtual variable and then studied the convergence rate of that virtual variable. On the contrary, our method directly studies the convergence rate for the original model parameter. Thus, the proof schema is totally different from (Yu et al., 2019a). In the following, we have the lemmas to bound the aforementioned difference. Their proof is deferred to Supplementary Materials.

Lemma 4.1. *Suppose Assumptions 3.1-3.3 hold, and* $0 < \alpha\eta < 1$*, we can get*

$$\begin{aligned}
&\mathbb{E}[\|m^{(k)}_{t+1} - \nabla F^{(k)}(x^{(k)}_t)\|^2] \\
&\le (1-\alpha\eta)\mathbb{E}[\|m^{(k)}_t - \nabla F^{(k)}(x^{(k)}_{t-1})\|^2] \\
&\quad + 2\alpha\eta C_g^2 L_f^2 \mathbb{E}[\|u^{(k)}_{t+1} - g^{(k)}(x^{(k)}_t)\|^2] \\
&\quad + \frac{2\eta\beta^2 L_F^2}{\alpha}\mathbb{E}[\|m^{(k)}_t\|^2] + 2\alpha^2\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) \,.
\end{aligned} \tag{11}$$

Lemma 4.2. *Suppose Assumptions 3.1-3.3 hold, and* $0 < \gamma\eta < 1$*, we can get*

$$\begin{aligned}
&\mathbb{E}[\|u^{(k)}_{t+1} - g^{(k)}(x^{(k)}_t)\|^2] \le \frac{\eta\beta^2 C_g^2}{\gamma}\mathbb{E}[\|m^{(k)}_t\|^2] + \gamma^2\eta^2\delta_g^2 \\
&\quad + (1-\gamma\eta)\mathbb{E}[\|u^{(k)}_t - g^{(k)}(x^{(k)}_{t-1})\|^2] \,.
\end{aligned} \tag{12}$$

Lemma 4.3. *Suppose Assumptions 3.1-3.3 hold, if* $p \le \frac{1}{4\beta\eta\sqrt{6C_f^2L_g^2+16C_g^4L_f^2}}$*, the consensus error* $\mathbb{E}[\|\bar{x}_t - x^{(k)}_t\|^2]$ *satisfies*

$$\begin{aligned}
&\sum_{t=0}^{T-1}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x^{(k)}_t\|^2] \le 256TKp^2\beta^2\eta^2C_g^2L_f^2\delta_g^2 \\
&+ 16TKp^2\beta^2\eta^2C_g^2\sigma_f^2 + 48TKp^2\beta^2\eta^2C_f^2\sigma_g^2 \,.
\end{aligned} \tag{13}$$

With above lemmas, we construct a novel potential function as follows:

$$P_t = \mathbb{E}[F(\bar{x}_t)] + \frac{\beta}{\alpha}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|m_{t+1}^{(k)} - \nabla F^{(k)}(x_t^{(k)})\|^2] + \frac{2\beta C_g^2 L_f^2}{\gamma}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] \,. \tag{14}$$

Based on this potential function, we can complete the proof of Theorem 3.4. Specifically, in the following, we demonstrate how the potential function evolves in each iteration.

Proof. Due to the smoothness of the loss function $F(\bar{x}_{t+1})$, we can get

$$\begin{aligned} F(\bar{x}_{t+1}) \leq\ & F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 - \frac{\beta\eta}{4}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\ & + \frac{\beta\eta}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(x_t^{(k)}) - m_{t+1}^{(k)}\|^2 \\ & + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\|\bar{x}_t - x_t^{(k)}\|^2 \,, \end{aligned} \tag{15}$$

where $L_F = C_g^2 L_f + C_f L_g$. Then, for the potential function, we can get

$$\begin{aligned} & P_{t+1} - P_t \\ \leq\ & -\frac{\beta\eta}{2}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] \\ & + \Big(\frac{2\eta\beta^3 C_g^4 L_f^2}{\gamma^2} + \frac{2\eta^2\beta^3 C_g^2 L_f^2 C_g^2}{\gamma} \\ & \quad + \frac{2\eta\beta^3 L_F^2}{\alpha^2} - \frac{\beta\eta}{4}\Big)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|m_{t+1}^{(k)}\|^2] \\ & + 2\beta\gamma\eta^2\delta_g^2 C_g^2 L_f^2 + 2\alpha\beta\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) \\ & + 2\beta\gamma^2\eta^3\delta_g^2 C_g^2 L_f^2 \,. \end{aligned} \tag{16}$$

By setting $\beta \leq \sqrt{\frac{1}{8}\Big/\Big(\frac{C_g^4 L_f^2}{\gamma^2} + \frac{C_g^4 L_f^2}{\gamma} + \frac{L_F^2}{\alpha^2}\Big)}$, we can see how the potential function evolves across iterations:

$$\begin{aligned} P_{t+1} - P_t \leq\ & -\frac{\beta\eta}{2}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] \\ & + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] + 2\beta\gamma\eta^2 C_g^2 L_f^2\delta_g^2 \\ & + 2\alpha\beta\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\beta\gamma^2\eta^3 C_g^2 L_f^2\delta_g^2 \,. \end{aligned} \tag{17}$$

Finally, by plugging Lemma 4.3 into this inequality, we complete the proof. □

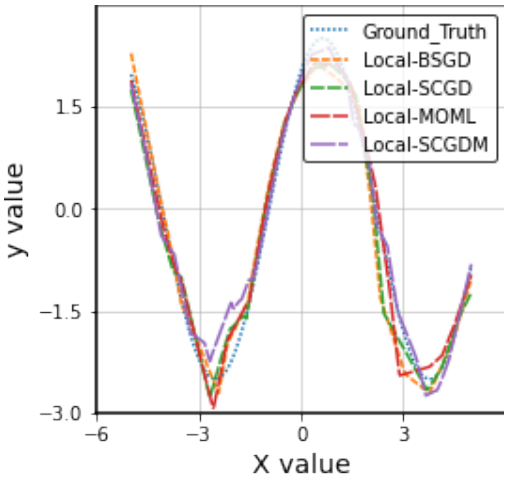

Figure 1. Fitted curves over an unseen task for our method Local-SCGDM and other baselines. The randomly selected ground truth sinusoid curve is $y = \frac{5}{2}sin(x + \frac{3}{2} * \frac{\pi}{5})$.

5. Numerical Experiments

In this section, we aim to evaluate the acceleration effect of the proposed algorithm Local-SCGDM with two Model-Agnostic Meta-Learning (MAML) tasks: the Sinewave Regression task and the Few-Shot Classification task over the Omniglot dataset. The formulation of the MAML tasks as compostional optimization problems are detailed in Eq. (8) All experiments are run over a machine with Intel Xeon Gold 6248 CPU and 4 Nvidia Tesla V100 GPUs. The code is written with Pytorch and the Federated Learning environment is simulated with Pytorch.distributed package.

5.1. Sinewave Regression

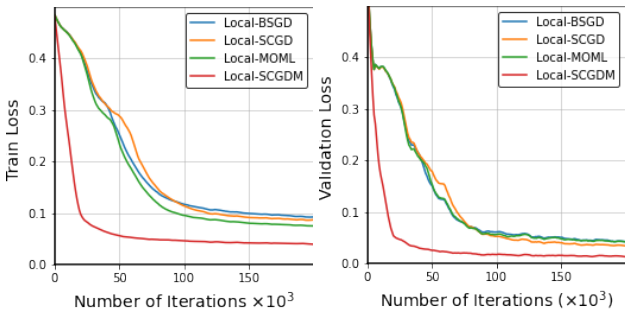

Figure 2. Train (Left) and Validation(Right) loss for our method and baselines.

In the first task, we evaluate our proposed algorithm Local-SCGDM over a 1-D sinusoid regression problem with various baselines: Local-BSGD (Huang et al., 2021) (i.e., Local-MAML (Finn et al., 2017)), Local-SCGD (Wang et al., 2017a), and Local-MOML (Wang et al., 2021). Note that

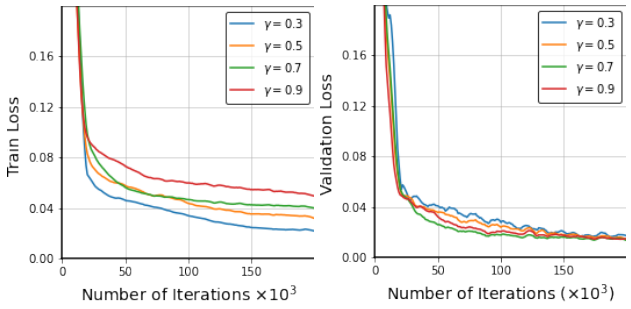

Figure 3. Different choices of inner state momentum coefficient γ. The left figure shows the train loss and the right figure shows the validation loss.

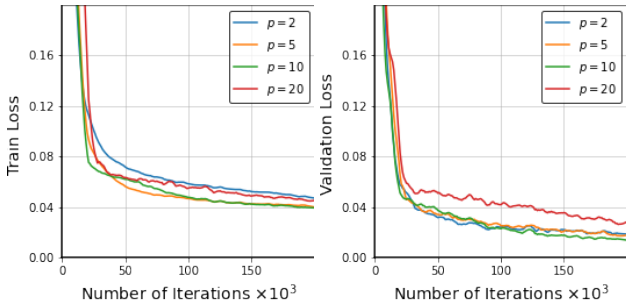

Figure 4. Different choices of the number of local epochs p. The left figure shows the training loss and the right figure shows the validation loss.

Local-SCGD is the federated version of SCGD (Wang et al., 2017a), where we average the model states and inner states u every p local iterations. The sinusoid regression task aims to fit a sinusoid function of the form $f(x) = Asin(x + \frac{b\pi}{5})$ where the amplitude A varies within $[0.1, 5]$ and the phase coefficient b varies within $[0, 5]$. This task can be effectively solved via meta learning while simple pre-training over all of the tasks leads to a degenerated all zero solution.

We follow a similar experimental protocol as in (Wang et al., 2021): we construct 25 different training tasks by choosing A = $\{1, 2, 3, 4, 5\}$ and b = $\{1, 2, 3, 4, 5\}$ and randomly and evenly distribute them over 5 clients. Then during training, we randomly sample 3 tasks for every client per meta-iteration. For each task we choose $K = 10$ samples of $x \in [-5, 5]$ randomly. At the test time, we sample 600 new tasks and for each task, the amplitude A and phase coefficient b are randomly chosen from the whole possible range. We use a two-layer fully-connected neural network with 40 hidden units and ReLU activation to perform training. The inner learning rate is 0.001 for all methods. For other hyper-parameters, we perform grid search for all methods and choose the setting with the best results. More precisely, for Local-BSGD (Local-MAML), we choose meta learning rate 0.01; for Local-SCGD, we choose meta learning rate 0.01 and the inner state momentum coefficient 0.9 (this algorithm diverges with smaller values); for Local-MOML, we choose meta learning rate 0.01, inner state momentum coefficient 0.7; for our Local-SCGDM, we choose η as 1, meta learning rate coefficient β as 0.01, meta momentum coefficient α as 0.8 and inner state momentum coefficient γ as 0.7. We set the number of local epochs as 5 in all comparison experiments. Finally, for fair comparison of our algorithm and Local-MOML, we also maintain a separate inner state u_i for each task i for our algorithm.

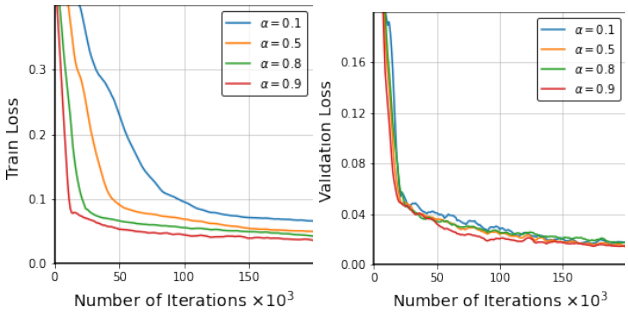

Figure 5. Different choices of momentum coefficient α. The left figure shows the training loss and the right figure shows the validation loss.

Training and test loss of our proposed algorithm and baselines are summarized in Figure 2. As shown in this figure, our local-SCGDM outperforms other baselines with a great margin. This validates the efficacy of using momentum to accelerate the convergence of MAML in the Federated Learning setting. Next, Figure 1 shows the fitted sinusoid of the meta-learned model over an unseen task. As shown in Figure 1, our algorithm fits the sinusoid curve well. Next, we present some ablation studies regarding some key hyper-parameters: inner state coefficient γ in Figure 3, the number of local epochs p in Figure 4 and the momentum coefficient α in Figure 5. As shown in Figure 3, smaller γ leads to faster fitting to the training tasks, however, it overfits the training data for very small γ, *e.g.*, $\gamma = 0.3$, and we get the best generalization performance at $\gamma = 0.7$. As for the number of local epochs, our Local-SCGDM performs well in different p values, and it gets both good train and validation performance when $p = 5$. For different choices of momentum coefficient α, our Local-SCGDM gets best performance both train and validation at value 0.9.

5.2. Few-shot Image Classification

Next, we evaluate our proposed Local-SCGDM with the few shot image classification task over the Omniglot dataset. The Omniglot dataset includes 1623 characters from 50 different alphabets and each character consists of 20 samples. We create the Federated version of the Omniglot dataset.

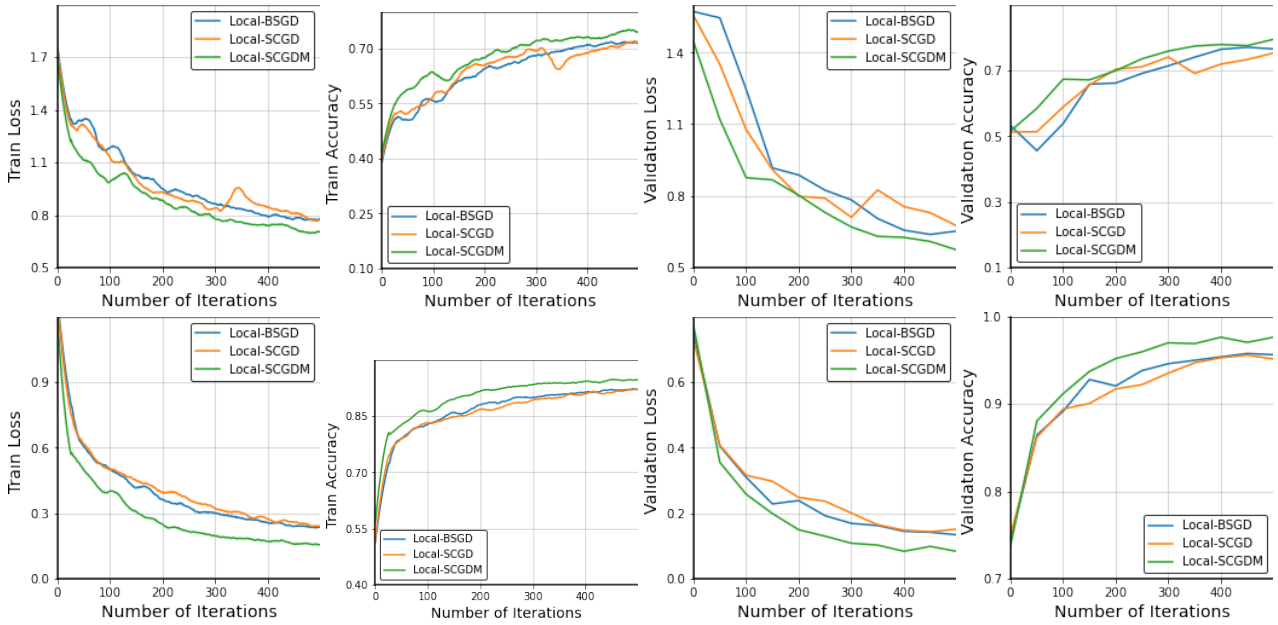

Figure 6. Few shot classification results over Omniglot Dataset. The top figures show results of the 5-way-1-shot case and the bottom figures show the 5-way-5-shot case. We smooth the curves for better visualization.

Firstly, we follow the experimental protocols of Vinyals et al. (2016) to divide the alphabets to train/validation/test with 33/5/12, respectively. Then we distribute one alphabet to a client, in other words, we consider 33 clients in experiments. As in the non-distributed setting, we perform N-way-K-shot classification, more specifically, for each task, we randomly sample N characters from the alphabet over that client and for each character, we sample K samples for training and 15 samples for validation. We augment the characters by performing rotation operations (multipliers of 90 degrees). Finally, we use a 4-layer convolutional neural network where each convolutional layer has 64 filters of 3×3 and it is followed by a batch-normalization layer (Finn et al., 2017).

In this task, we compare our algorithm with baselines Local-BSGD (Huang et al., 2021) (i.e., Local-MAML (Finn et al., 2017)) and Local-SCGD (Wang et al., 2017a). MOML is ignored due to its requirement of keeping an inner state for each task which is prohibitive due to large number of tasks in the Omniglot dataset. For all methods, the inner learning rate is set as 0.4, for other hyper-parameters, we perform grid search for each method and choose the setting with best results. Different sets of hyper-parameters are used for different cases, *e.g.*, for the 5-way-1-shot case, for Local-BSGD (Local-MAML), we choose meta learning rate 0.1; for Local-SCGD, we choose meta learning rate 0.1 and inner state momentum coefficient 0.99; for our Local-SCGDM, we choose η as 1, meta learning rate coefficient β as 0.1, meta momentum coefficient α as 0.9 and inner state momentum coefficient γ as 0.99.

The experimental results are summarized in Figure 6, which includes the 5-way-1-shot and 5-way-5-shot cases, while the 20-way-1-shot and 20-way-5-shot cases are included in Supplementary Materials. As shown in these figures, our algorithm outperforms baselines with a great margin for both training loss and validation accuracy. This confirms that our algorithm can effectively accelerate SCGD by using momentum in federated learning.

6. Conclusion

In this paper, we proposed a novel local stochastic compositional gradient descent with momentum method for federated compositional optimization problems. By introducing the momentum, our method can improve the sample complexity significantly compared with existing methods. Specifically, our method can achieve $O(1/\epsilon^4)$ sample complexity and $O(1/\epsilon^3)$ communication complexity. To the best of our knowledge, this is the first method achieving such kinds of results. Meanwhile, we proposed a novel theoretical analysis strategy to establish the convergence rate. In particular, we developed a novel potential function and studied how this potential function evolves across iterations for establishing the convergence rate. Finally, we use our method to optimize the federated model-agnostic meta-learning problem. The extensive experimental results on benchmark datasets confirm the efficacy of our method.

Acknowledgement

Junyi Li and Heng Huang were partially supported by NSF IIS 1845666, 1852606, 1838627, 1837956, 1956002, IIA 2040588.

References

Basu, D., Data, D., Karakus, C., and Diggavi, S. Qsparse-local-sgd: Distributed sgd with quantization, sparsification, and local computations. *arXiv preprint arXiv:1906.02367*, 2019.

Chen, T., Sun, Y., and Yin, W. Solving stochastic compositional optimization is nearly as easy as solving stochastic optimization. *arXiv preprint arXiv:2008.10847*, 2020.

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

Chen, T., Sun, Y., and Yin, W. Tighter analysis of alternating stochastic gradient method for stochastic nested problems. *arXiv preprint arXiv:2106.13781*, 2021b.

Cutkosky, A. and Orabona, F. Momentum-based variance reduction in non-convex sgd. In *Advances in Neural Information Processing Systems*, pp. 15236–15245, 2019.

Das, R., Hashemi, A., Sanghavi, S., and Dhillon, I. S. Improved convergence rates for non-convex federated learning with compression. *arXiv e-prints*, pp. arXiv–2012, 2020.

Fang, C., Li, C. J., Lin, Z., and Zhang, T. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pp. 689–699, 2018.

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

Gao, H. and Huang, H. Periodic stochastic gradient descent with momentum for decentralized training. *arXiv preprint arXiv:2008.10435*, 2020.

Gao, H. and Huang, H. Fast training method for stochastic compositional optimization problems. *Advances in Neural Information Processing Systems*, 34, 2021.

Gao, H., Xu, A., and Huang, H. On the convergence of communication-efficient local sgd for federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7510–7518, 2021.

Ghadimi, S., Ruszczynski, A., and Wang, M. A single timescale stochastic approximation method for nested stochastic optimization. *SIAM Journal on Optimization*, 30(1):960–979, 2020.

Huang, F., Li, J., and Huang, H. Compositional federated learning: Applications in distributionally robust averaging and meta learning. *arXiv preprint arXiv:2106.11264*, 2021.

Ji, K., Lee, J. D., Liang, Y., and Poor, H. V. Convergence of meta-learning with task-specific adaptation over partial parameters. *arXiv preprint arXiv:2006.09486*, 2020.

Karimireddy, S. P., Jaggi, M., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Suresh, A. T. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020a.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. SCAFFOLD: Stochastic controlled averaging for federated learning. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 13–18 Jul 2020b. URL https://proceedings.mlr.press/v119/karimireddy20a.html.

Khanduri, P., Sharma, P., Yang, H., Hong, M., Liu, J., Rajawat, K., and Varshney, P. K. Stem: A stochastic two-sided momentum algorithm achieving near-optimal sample and communication complexities for federated learning. *arXiv preprint arXiv:2106.10435*, 2021.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

Liu, W., Chen, L., Chen, Y., and Zhang, W. Accelerating federated learning via momentum gradient descent. *IEEE Transactions on Parallel and Distributed Systems*, 31(8): 1754–1766, 2020.

Murata, T. and Suzuki, T. Bias-variance reduced local sgd for less heterogeneous federated learning. *arXiv preprint arXiv:2102.03198*, 2021.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takáč, M. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pp. 2613–2621. PMLR, 2017.

Qi, Q., Guo, Z., Xu, Y., Jin, R., and Yang, T. An online method for distributionally deep robust optimization. *arXiv preprint arXiv:2006.10138*, 2020.

Qi, Q., Luo, Y., Xu, Z., Ji, S., and Yang, T. Stochastic optimization of area under precision-recall curve for deep learning with provable convergence. *arXiv preprint arXiv:2104.08736*, 2021.

Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., and Pedarsani, R. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *International Conference on Artificial Intelligence and Statistics*, pp. 2021–2031. PMLR, 2020.

Rothchild, D., Panda, A., Ullah, E., Ivkin, N., Stoica, I., Braverman, V., Gonzalez, J., and Arora, R. Fetchsgd: Communication-efficient federated learning with sketching. In *International Conference on Machine Learning*, pp. 8253–8265. PMLR, 2020.

Stich, S. U. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29:3630–3638, 2016.

Wang, B., Yuan, Z., Ying, Y., and Yang, T. Memory-based optimization methods for model-agnostic meta-learning. *arXiv preprint arXiv:2106.04911*, 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. *The Journal of Machine Learning Research*, 18(1):3721–3743, 2017b.

Xu, J., Wang, S., Wang, L., and Yao, A. C.-C. Fedcm: Federated learning with client-level momentum. *arXiv preprint arXiv:2106.10874*, 2021.

Yang, J. and Hu, W. Stochastic recursive momentum method for non-convex compositional optimization. *arXiv preprint arXiv:2006.01688*, 2020.

Yu, H., Jin, R., and Yang, S. On the linear speedup analysis of communication efficient momentum sgd for distributed non-convex optimization. *arXiv preprint arXiv:1905.03817*, 2019a.

Yu, H., Yang, S., and Zhu, S. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5693–5700, 2019b.

Yuan, H. and Hu, W. Stochastic recursive momentum method for non-convex compositional optimization. *arXiv preprint arXiv:2006.01688*, 2020.

Yuan, H., Lian, X., and Liu, J. Stochastic recursive variance reduction for efficient smooth non-convex compositional optimization. *arXiv preprint arXiv:1912.13515*, 2019.

Zhang, J. and Xiao, L. A composite randomized incremental gradient method. In *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*, pp. 9078–9088, 2019b.

A. Supplementary Materials

A.1. More Experiments

In this subsection, we show more experimental results for the federated few-shot classification task over the Omniglot Dataset. We show results of the 20-way-1-shot and 20-way-5-shot cases in Figure 7.

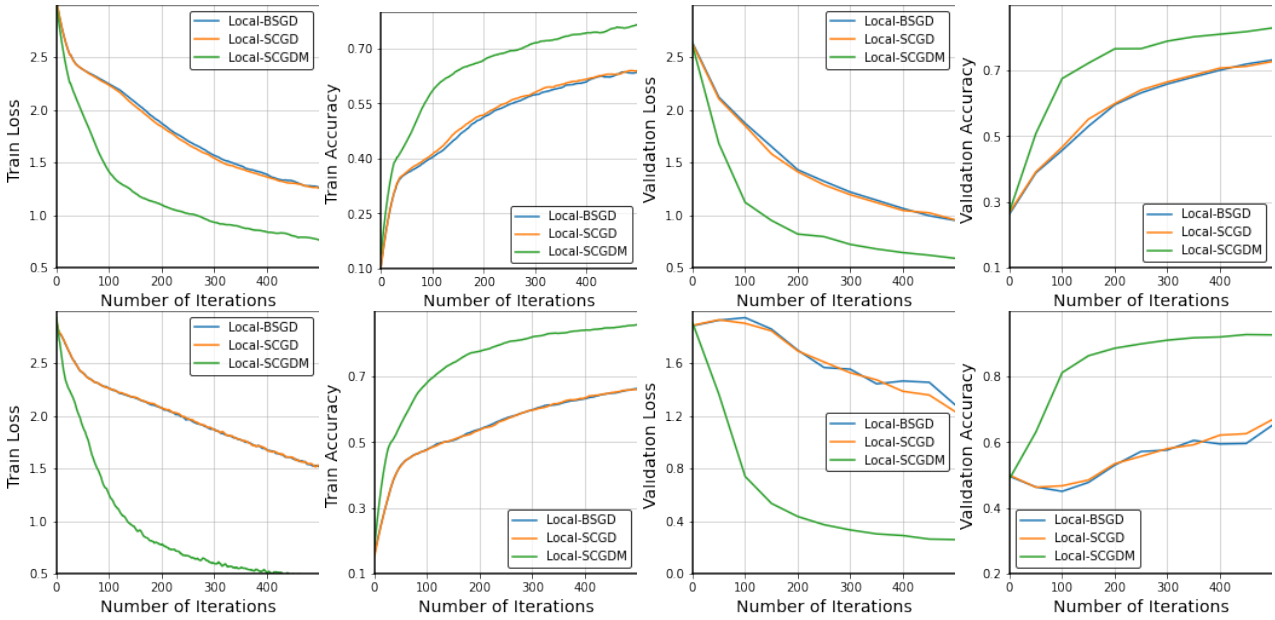

Figure 7. Few shot classification results over Omniglot Dataset. The top figures show results of 20-way-1-shot and the bottom figures show the 20-way-5-shot case. We smooth the curves for better visualization.

A.2. Proof of Lemma 3.9

Proof. From the definition of $g^{(k)}$, we can get

$$\begin{aligned}
&\|\nabla g^{(k)}(x) - \nabla g^{(k)}(y)\| \\
&= \|I - \lambda\nabla^2\mathcal{L}_i^{(k)}(x) - I + \lambda\nabla^2\mathcal{L}_i^{(k)}(y)\| \\
&= \lambda\|\nabla^2\mathcal{L}_i^{(k)}(x) - \nabla^2\mathcal{L}_i^{(k)}(y)\| \\
&\leq \lambda L_2\|x - y\| \,.
\end{aligned} \tag{18}$$

Thus, $L_g = \lambda L_2$. From the definition of $\nabla f^{(k)}$, we can get

$$\|\nabla f^{(k)}(x) - \nabla f^{(k)}(y)\| = \|\nabla\mathcal{L}_i^{(k)}(x) - \nabla\mathcal{L}_i^{(k)}(y)\| \leq L_1\|x - y\| \,. \tag{19}$$

Thus, $L_f = L_1$. Moreover, due to

$$\|\nabla g^{(k)}(x)\| = \|I - \lambda\nabla^2\mathcal{L}_i^{(k)}(x)\| \leq \|I\| + \|\lambda\nabla^2\mathcal{L}_i^{(k)}(x)\| \leq 1 + \lambda L_1 \,, \tag{20}$$

we can get

$$\begin{aligned}
&\mathbb{E}[\|\nabla g^{(k)}(x;\xi)\|^2] = \mathbb{E}[\|I - \lambda\nabla^2\mathcal{L}_i^{(k)}(x;\xi)\|^2] \\
&\leq \mathbb{E}[\|I - \lambda\nabla^2\mathcal{L}_i^{(k)}(x) + \lambda\nabla^2\mathcal{L}_i^{(k)}(x) - \lambda\nabla^2\mathcal{L}_i^{(k)}(x;\xi)\|^2] \\
&\leq \|I - \lambda\nabla^2\mathcal{L}_i^{(k)}(x)\|^2 + \mathbb{E}[\|\lambda\nabla^2\mathcal{L}_i^{(k)}(x) - \lambda\nabla^2\mathcal{L}_i^{(k)}(x;\xi)\|^2] \\
&\leq (1 + \lambda L_1)^2 + \lambda^2\sigma_2^2 \,.
\end{aligned} \tag{21}$$

Thus, $C_g^2 = (1+\lambda L_1)^2 + \lambda^2\sigma_2^2$. In addition, due to $\|\nabla \mathcal{L}_i^{(k)}(y)\| \leq G$, we can get

$$\begin{aligned}
&\mathbb{E}[\|\nabla f^{(k)}(y;\zeta)\|^2] \\
&= \mathbb{E}[\|\nabla \mathcal{L}_i^{(k)}(y;\zeta) - \nabla \mathcal{L}_i^{(k)}(y) + \nabla \mathcal{L}_i^{(k)}(y)\|^2] \\
&= \mathbb{E}[\|\nabla \mathcal{L}_i^{(k)}(y;\zeta) - \nabla \mathcal{L}_i^{(k)}(y)\|^2] + \|\nabla \mathcal{L}_i^{(k)}(y)\|^2 \\
&\leq \sigma_1^2 + G^2 \ .
\end{aligned} \tag{22}$$

Thus, $C_f^2 = \sigma_1^2 + G^2$. As for the variance, we can get

$$\begin{aligned}
&\mathbb{E}[\|\nabla f^{(k)}(y;\zeta) - \nabla f^{(k)}(y)\|^2] \\
&= \mathbb{E}[\|\nabla \mathcal{L}_i^{(k)}(y;\zeta) - \nabla \mathcal{L}_i^{(k)}(y)\|^2] \\
&\leq \sigma_1^2 = \sigma_f^2 \ ,
\end{aligned}$$

$$\begin{aligned}
&\mathbb{E}[\|\nabla g^{(k)}(x;\xi) - \nabla g^{(k)}(x)\|^2] \\
&= \mathbb{E}[\|I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x;\xi) - I + \lambda\nabla^2 \mathcal{L}_i^{(k)}(x)\|^2] \\
&\leq \lambda^2 \mathbb{E}[\|\nabla^2 \mathcal{L}_i^{(k)}(x;\xi) - \nabla^2 \mathcal{L}_i^{(k)}(x)\|^2] \\
&\leq \lambda^2\sigma_2^2 = \sigma_g^2 \ ,
\end{aligned} \tag{23}$$

$$\begin{aligned}
&\mathbb{E}[\|g^{(k)}(x;\xi) - g^{(k)}(x)\|^2] \\
&= \mathbb{E}[|I - \lambda\nabla \mathcal{L}_i^{(k)}(x;\xi) - I + \lambda\nabla \mathcal{L}_i^{(k)}(x)\|^2] \\
&\leq \lambda^2 \mathbb{E}[\|\nabla \mathcal{L}_i^{(k)}(x;\xi) - \nabla \mathcal{L}_i^{(k)}(x)\|^2] \\
&\leq \lambda^2\sigma_1^2 = \delta_g^2 \ .
\end{aligned}$$

Finally, as for the smoothness of $F^{(k)}$, we can get

$$\begin{aligned}
&\|\nabla F^{(k)}(x) - \nabla F^{(k)}(y)\| \\
&= \|(I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(x - \lambda\nabla \mathcal{L}_i^{(k)}(x)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(y))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y))\| \\
&= \|(I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(x - \lambda\nabla \mathcal{L}_i^{(k)}(x)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y)) \\
&\quad + (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(y))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y))\| \\
&\leq \|(I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(x - \lambda\nabla \mathcal{L}_i^{(k)}(x)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y))\| \\
&\quad + \|(I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(y))\nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y))\| \\
&\leq (1+\lambda L_1)\|\nabla \mathcal{L}_i^{(k)}(x - \lambda\nabla \mathcal{L}_i^{(k)}(x)) - \nabla \mathcal{L}_i^{(k)}(y - \lambda\nabla \mathcal{L}_i^{(k)}(y))\| + G\|(I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(x)) - (I - \lambda\nabla^2 \mathcal{L}_i^{(k)}(y))\| \\
&\leq (1+\lambda L_1)L_1\|x - \lambda\nabla \mathcal{L}_i^{(k)}(x) - y + \lambda\nabla \mathcal{L}_i^{(k)}(y)\| + \lambda G\|\nabla^2 \mathcal{L}_i^{(k)}(x) - \nabla^2 \mathcal{L}_i^{(k)}(y)\| \\
&\leq (1+\lambda L_1)L_1(\|x-y\| + \lambda L_1\|x-y\|) + \lambda G L_2\|x-y\| \\
&= ((1+\lambda L_1)^2 L_1 + \lambda G L_2)\|x-y\| \ .
\end{aligned} \tag{24}$$

□

A.3. Proof of Lemma 4.1

Proof. Denoting $v_t^{(k)} = \nabla g^{(k)}(x_t^{(k)}; \xi_t^{(k)})$, one can get

$$
\begin{aligned}
&\mathbb{E}[\|m_{t+1}^{(k)} - \nabla F^{(k)}(x_t^{(k)})\|^2] \\
&= \mathbb{E}[\|(1-\alpha\eta)(m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})) + (1-\alpha\eta)(\nabla F^{(k)}(x_{t-1}^{(k)}) - \nabla F^{(k)}(x_t^{(k)})) \\
&\quad + \alpha\eta(z_t^{(k)} - \nabla F^{(k)}(x_t^{(k)}))\|^2] \\
&= \mathbb{E}[\|(1-\alpha\eta)(m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})) + (1-\alpha\eta)(\nabla F^{(k)}(x_{t-1}^{(k)}) - \nabla F^{(k)}(x_t^{(k)})) \\
&\quad + \alpha\eta((v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}; \zeta_t^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) + (v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})) \\
&\quad + (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})) - \nabla g^{(k)}(x_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})))\|^2] \\
&= \mathbb{E}[\|(1-\alpha\eta)(m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})) + (1-\alpha\eta)(\nabla F^{(k)}(x_{t-1}^{(k)}) - \nabla F^{(k)}(x_t^{(k)})) \\
&\qquad + \alpha\eta((v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})))\|^2] \\
&\quad + \alpha^2\eta^2 \mathbb{E}[\|(v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}; \zeta_t^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) \\
&\qquad + (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})) - \nabla g^{(k)}(x_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)}))\|^2] \\
&\le (1+a)(1-\alpha\eta)^2 \mathbb{E}[\|m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})\|^2] + (1+\frac{1}{a})\mathbb{E}[\|(1-\alpha\eta)(\nabla F^{(k)}(x_{t-1}^{(k)}) - \nabla F^{(k)}(x_t^{(k)})) \\
&\qquad + \alpha\eta((v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})))\|^2] \\
&\quad + \alpha^2\eta^2 \mathbb{E}[\|(v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}; \zeta_t^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) \\
&\qquad + (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})) - \nabla g^{(k)}(x_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)}))\|^2] \\
&\le (1-\alpha\eta)\mathbb{E}[\|m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{2(1-\alpha\eta)^2}{\alpha\eta}\mathbb{E}[\|\nabla F^{(k)}(x_{t-1}^{(k)}) - \nabla F^{(k)}(x_t^{(k)})\|^2] \\
&\quad + 2\alpha\eta\mathbb{E}[\|(v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)}))\|^2] \\
&\quad + \alpha^2\eta^2 \mathbb{E}[\|(v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}; \zeta_t^{(k)}) - (v_t^{(k)})^T \nabla_g f^{(k)}(u_{t+1}^{(k)}) \\
&\qquad + (v_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)})) - \nabla g^{(k)}(x_t^{(k)})^T \nabla_g f^{(k)}(g^{(k)}(x_t^{(k)}))\|^2] \\
&\le (1-\alpha\eta)\mathbb{E}[\|m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{2L_F^2}{\alpha\eta}\mathbb{E}[\|x_t^{(k)} - x_{t-1}^{(k)}\|^2] \\
&\quad + 2\alpha\eta C_g^2 L_f^2 \mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] + 2\alpha^2\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) \\
&\le (1-\alpha\eta)\mathbb{E}[\|m_t^{(k)} - \nabla F^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{2\eta\beta^2 L_F^2}{\alpha}\mathbb{E}[\|m_t^{(k)}\|^2] \\
&\quad + 2\alpha\eta C_g^2 L_f^2 \mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] + 2\alpha^2\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2)\,,
\end{aligned}
\tag{25}
$$

where the fourth equality follows from $\mathbb{E}[f^{(k)}(u_{t+1}^{(k)}; \zeta_t^{(k)})] = f^{(k)}(u_{t+1}^{(k)})$ and $\mathbb{E}[v_t^{(k)}] = \nabla g^{(k)}(x_t^{(k)})$, the fifth equality follows from $a = \frac{\alpha\eta}{1-\alpha\eta}$, the second to last step follows from $\alpha\eta < 1$ and Assumptions 3.1-3.3.

□

A.4. Proof of Lemma 4.2

Proof. Based on the updating rule of $u_{t+1}^{(k)}$, one can get

$$
\begin{aligned}
&\mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] \\
&= \mathbb{E}[\|(1-\gamma\eta)u_t^{(k)} + \gamma\eta g^{(k)}(x_t^{(k)};\xi^{(k)}) - g^{(k)}(x_t^{(k)})\|^2] \\
&= \mathbb{E}[\|(1-\gamma\eta)(u_t^{(k)} - g^{(k)}(x_t^{(k)})) + \gamma\eta(g^{(k)}(x_t^{(k)};\xi^{(k)}) - g^{(k)}(x_t^{(k)}))\|^2] \\
&= \mathbb{E}[\|(1-\gamma\eta)(u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)}) + g^{(k)}(x_{t-1}^{(k)}) - g^{(k)}(x_t^{(k)})) + \gamma\eta(g^{(k)}(x_t^{(k)};\xi^{(k)}) - g^{(k)}(x_t^{(k)}))\|^2] \\
&\le (1-\gamma\eta)^2\mathbb{E}[\|u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)}) + g^{(k)}(x_{t-1}^{(k)}) - g^{(k)}(x_t^{(k)})\|^2] + \gamma^2\eta^2\delta_g^2 \\
&\le (1-\gamma\eta)^2(1+\frac{1}{a})\mathbb{E}[\|u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)})\|^2] + (1-\gamma\eta)^2(1+a)\mathbb{E}[\|g^{(k)}(x_{t-1}^{(k)}) - g^{(k)}(x_t^{(k)})\|^2] + \gamma^2\eta^2\delta_g^2 \\
&= (1-\gamma\eta)\mathbb{E}[\|u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{(1-\gamma\eta)^2}{\gamma\eta}\mathbb{E}[\|g^{(k)}(x_{t-1}^{(k)}) - g^{(k)}(x_t^{(k)})\|^2] + \gamma^2\eta^2\delta_g^2 \\
&\le (1-\gamma\eta)\mathbb{E}[\|u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{C_g^2}{\gamma\eta}\mathbb{E}[\|x_t^{(k)} - x_{t-1}^{(k)}\|^2] + \gamma^2\eta^2\delta_g^2 \\
&\le (1-\gamma\eta)\mathbb{E}[\|u_t^{(k)} - g^{(k)}(x_{t-1}^{(k)})\|^2] + \frac{\eta\beta^2 C_g^2}{\gamma}\mathbb{E}[\|m_t^{(k)}\|^2] + \gamma^2\eta^2\delta_g^2\,,
\end{aligned}
\tag{26}
$$

where the fourth step follows from $\mathbb{E}[g^{(k)}(x_t^{(k)};\xi_t^{(k)})] = g^{(k)}(x_t^{(k)})$ and Assumption 3.3, the sixth step follows from $a = \frac{1-\gamma\eta}{\gamma\eta}$, the second to last step follows from $\gamma\eta < 1$ and Assumption 3.2.

□

A.5. Proof of Lemma 4.3

Lemma A.1. *Suppose Assumptions 3.1-3.3 hold, the consensus error $\|u_t^{(k)} - \bar{u}_t\|^2$ satisfies*

$$
\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|u_{t'}^{(k)} - \bar{u}_{t'}\|^2] \le 8pK\delta_g^2 + 8C_g^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)} - \bar{x}_{t'}\|^2]\,,
\tag{27}
$$

where $s_t = \lfloor\frac{t}{p}\rfloor$.

Proof. Based on Algorithm 1, one can get

$$
\begin{aligned}
&\sum_{k=1}^{K}\|u_t^{(k)} - \bar{u}_t\|^2 \\
&= \sum_{k=1}^{K}\|(1-\gamma\eta)^{t-s_tp}u_{s_tp}^{(k)} + \gamma\eta\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}g^{(k)}(x_{t'}^{(k)};\xi^{(k)}) \\
&\quad - \Big((1-\gamma\eta)^{t-s_tp}\bar{u}_{s_tp} + \gamma\eta\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}\frac{1}{K}\sum_{k'=1}^{K}g^{(k)}(x_{t'}^{(k')};\xi^{(k')})\Big)\|^2 \\
&= \gamma^2\eta^2\sum_{k=1}^{K}\|\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}\Big(g^{(k)}(x_{t'}^{(k)};\xi^{(k)}) - \frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\Big)\|^2\,,
\end{aligned}
\tag{28}
$$

where the last step holds due to $u_{s_tp}^{(k)} = \bar{u}_{s_tp}$. Define $w_t \triangleq \sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'} = \sum_{t'=0}^{t-1-s_tp}(1-\gamma\eta)^{t'} = \frac{1-(1-\gamma\eta)^{t-s_tp}}{\gamma\eta}$,

then one can get

$$\begin{aligned}
&\sum_{k=1}^{K}\|\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}\Big(g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k)}(x_{t'}^{(k')};\xi^{(k')})\Big)\|^2\\
&=w_t^2\sum_{k=1}^{K}\|\sum_{t'=s_tp}^{t-1}\frac{(1-\gamma\eta)^{t-1-t'}}{w_t}\Big(g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\Big)\|^2\\
&\leq w_t^2\sum_{k=1}^{K}\sum_{t'=s_tp}^{t-1}\frac{(1-\gamma\eta)^{t-1-t'}}{w_t}\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2\\
&\leq\frac{1}{\gamma\eta}\sum_{k=1}^{K}\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2\,.
\end{aligned}\tag{29}$$

As a result, one can get

$$\sum_{k=1}^{K}\mathbb{E}[\|u_t^{(k)}-\bar{u}_t\|^2]\leq\gamma\eta\sum_{k=1}^{K}\sum_{t'=s_tp}^{t-1}(1-\gamma\eta)^{t-1-t'}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\,.\tag{30}$$

Furthermore, one can bound

$$\begin{aligned}
&\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\\
&=\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-g^{(k)}(x_{t'}^{(k)})+g^{(k)}(x_{t'}^{(k)})-g^{(k)}(\bar{x}_{t'})\\
&\quad+\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(\bar{x}_{t'})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')})+\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\\
&\leq 4\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-g^{(k)}(x_{t'}^{(k)})\|^2]+4\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)})-g^{(k)}(\bar{x}_{t'})\|^2]\\
&\quad+4\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(\bar{x}_{t'})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')})\|^2]\\
&\quad+4\sum_{k=1}^{K}\mathbb{E}[\|\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\\
&\leq 8K\delta_g^2+4C_g^2\sum_{k=1}^{K}\|x_{t'}^{(k)}-\bar{x}_{t'}\|^2+4C_g^2\sum_{k=1}^{K}\frac{1}{K}\sum_{k'=1}^{K}\mathbb{E}[\|x_{t'}^{(k')}-\bar{x}_{t'}\|^2]\\
&=8K\delta_g^2+8C_g^2\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)}-\bar{x}_{t'}\|^2]\,.
\end{aligned}\tag{31}$$

By combining above two inequalities, one can get

$$
\begin{aligned}
&\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|u_{t'}^{(k)}-\bar{u}_{t'}\|^2]\\
&\leq \gamma\eta\sum_{t'=s_tp}^{t-1}\sum_{t''=s_tp}^{t'-1}(1-\gamma\eta)^{t'-1-t''}\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t''}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t''}^{(k')};\xi^{(k')})\|^2]\\
&=\gamma\eta\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\sum_{t''=0}^{t-t'-1}(1-\gamma\eta)^{t''} \qquad (32)\\
&\leq\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|g^{(k)}(x_{t'}^{(k)};\xi^{(k)})-\frac{1}{K}\sum_{k'=1}^{K}g^{(k')}(x_{t'}^{(k')};\xi^{(k')})\|^2]\\
&\leq 8pK\delta_g^2+8C_g^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)}-\bar{x}_{t'}\|^2]\,.
\end{aligned}
$$

□

Lemma A.2. *(Lemma 5 (Gao & Huang, 2021)) Suppose Assumptions 3.1-3.3 hold, the consensus error* $\|z_{t+1}^{(k)}-\bar{z}_{t+1}\|^2$ *satisfies*

$$
\sum_{k=1}^{K}\|z_{t+1}^{(k)}-\bar{z}_{t+1}\|^2\leq 48C_f^2L_g^2\sum_{k=1}^{K}\|x_{t+1}^{(k)}-\bar{x}_{t+1}\|^2+16C_g^2L_f^2\sum_{k=1}^{K}\|u_{t+1}^{(k)}-\bar{u}_{t+1}\|^2+8KC_g^2\sigma_f^2+24KC_f^2\sigma_g^2\,. \qquad (33)
$$

Based on Lemma A.1 and Lemma A.2, we are ready to prove Lemma 4.3.

Proof. Denoting $s_t=\lfloor\frac{t}{p}\rfloor$, based on Algorithm 1, one can get

$$
\begin{aligned}
&m_t^{(k)}-\bar{m}_t\\
&=\sum_{t'=s_tp}^{t-1}(1-\alpha\eta)^{t'-s_tp}m_{s_tp}^{(k)}+\alpha\eta\sum_{t'=s_tp}^{t-1}(1-\alpha\eta)^{t-1-t'}z_{t'}^{(k)}\\
&\quad-\sum_{t'=s_tp}^{t-1}(1-\alpha\eta)^{t'-s_tp}\frac{1}{K}\sum_{k'=1}^{K}m_{s_tp}^{(k')}-\alpha\eta\sum_{t'=s_tp}^{t-1}(1-\alpha\eta)^{t-1-t'}\bar{z}_{t'} \qquad (34)\\
&=\alpha\eta\sum_{t'=s_tp}^{t-1}(1-\alpha\eta)^{t-1-t'}\left(z_{t'}^{(k)}-\bar{z}_{t'}\right).
\end{aligned}
$$

Then, for the consensus error $\|\bar{x}_t - x_t^{(k)}\|^2$, one can get

$$
\begin{aligned}
&\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] \\
&= \sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_{s_tp} - \beta\eta\sum_{t'=s_tp}^{t-1}\bar{m}_{t'+1} - (x_{s_tp}^{(k)} - \beta\eta\sum_{t'=s_tp}^{t-1}m_{t'+1}^{(k)})\|^2] \\
&= \beta^2\eta^2\sum_{k=1}^{K}\mathbb{E}[\|\sum_{t'=s_tp}^{t-1}(m_{t'+1}^{(k)} - \bar{m}_{t'+1})\|^2] \\
&= \alpha^2\beta^2\eta^4\sum_{k=1}^{K}\mathbb{E}[\|\sum_{t'=s_tp}^{t-1}\sum_{t''=s_tp}^{t'}(1-\alpha\eta)^{t'-t''}\left(z_{t''}^{(k)} - \bar{z}_{t''}\right)\|^2] \\
&= \alpha^2\beta^2\eta^4\sum_{k=1}^{K}\mathbb{E}[\|\sum_{t'=s_tp}^{t-1}\left(z_{t'}^{(k)} - \bar{z}_{t'}\right)\sum_{t''=0}^{t-t'}(1-\alpha\eta)^{t''}\|^2] \\
&\le p\beta^2\eta^2\sum_{k=1}^{K}\sum_{t'=s_tp}^{t-1}\mathbb{E}[\|z_{t'}^{(k)} - \bar{z}_{t'}\|^2] \\
&\le 48p\beta^2\eta^2C_f^2L_g^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)} - \bar{x}_{t'}\|^2] + 16p\beta^2\eta^2C_g^2L_f^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|u_{t'}^{(k)} - \bar{u}_{t'}\|^2] \\
&\quad + 8Kp^2\beta^2\eta^2C_g^2\sigma_f^2 + 24Kp^2\beta^2\eta^2C_f^2\sigma_g^2 \\
&\le 48p\beta^2\eta^2C_f^2L_g^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)} - \bar{x}_{t'}\|^2] + 128p\beta^2\eta^2C_g^4L_f^2\sum_{t'=s_tp}^{t-1}\sum_{k=1}^{K}\mathbb{E}[\|x_{t'}^{(k)} - \bar{x}_{t'}\|^2] \\
&\quad + 128Kp^2\beta^2\eta^2C_g^2L_f^2\delta_g^2 + 8Kp^2\beta^2\eta^2C_g^2\sigma_f^2 + 24Kp^2\beta^2\eta^2C_f^2\sigma_g^2\,,
\end{aligned}
\tag{35}
$$

where the second to last step holds due to Lemma A.2, the last step holds due to Lemma A.1. By summing t from 0 to $T-1$, one can get

$$
\begin{aligned}
\sum_{t=0}^{T-1}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] &\le (48p^2\beta^2\eta^2C_f^2L_g^2 + 128p^2\beta^2\eta^2C_g^4L_f^2)\sum_{t=0}^{T-1}\sum_{k=1}^{K}\mathbb{E}[\|x_t^{(k)} - \bar{x}_t\|^2] \\
&\quad + 128TKp^2\beta^2\eta^2C_g^2L_f^2\delta_g^2 + 8TKp^2\beta^2\eta^2C_g^2\sigma_f^2 + 24TKp^2\beta^2\eta^2C_f^2\sigma_g^2\,.
\end{aligned}
\tag{36}
$$

By setting $p \le \frac{1}{4\beta\eta\sqrt{6C_f^2L_g^2+16C_g^4L_f^2}}$, one can get

$$
\sum_{t=0}^{T-1}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] \le 256TKp^2\beta^2\eta^2C_g^2L_f^2\delta_g^2 + 16TKp^2\beta^2\eta^2C_g^2\sigma_f^2 + 48TKp^2\beta^2\eta^2C_f^2\sigma_g^2\,. \tag{37}
$$

□

A.6. Proof of Theorem 3.4

Based on aforementioned lemmas, we are ready to prove the convergence of Theorem 3.4.

Proof. Due to the smoothness of $F(x)$, one can get

$$
\begin{aligned}
&F(\bar{x}_{t+1}) \leq F(\bar{x}_t) + \langle \nabla F(\bar{x}_t), \bar{x}_{t+1} - \bar{x}_t \rangle + \frac{L_F}{2}\|\bar{x}_{t+1} - \bar{x}_t\|^2 \\
&= F(\bar{x}_t) - \beta\eta \langle \nabla F(\bar{x}_t), \frac{1}{K}\sum_{k=1}^{K} m_{t+1}^{(k)} \rangle + \frac{\beta^2\eta^2 L_F}{2}\|\frac{1}{K}\sum_{k=1}^{K} m_{t+1}^{(k)}\|^2 \\
&\leq F(\bar{x}_t) - \beta\eta \frac{1}{K}\sum_{k=1}^{K} \langle \nabla F(\bar{x}_t), m_{t+1}^{(k)} \rangle + \frac{\beta^2\eta^2 L_F}{2}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\
&\leq F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 - \frac{\beta\eta}{2}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 + \frac{\beta\eta}{2}\frac{1}{K}\sum_{k=1}^{K}\|\nabla F(\bar{x}_t) - m_{t+1}^{(k)}\|^2 + \frac{\beta^2\eta^2 L_F}{2}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\
&= F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 + \frac{\beta\eta}{2}\frac{1}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(\bar{x}_t) - m_{t+1}^{(k)}\|^2 + \Big(\frac{\beta^2\eta^2 L_F}{2} - \frac{\beta\eta}{2}\Big)\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\
&\leq F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 - \frac{\beta\eta}{4}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 + \frac{\beta\eta}{2}\frac{1}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(\bar{x}_t) - m_{t+1}^{(k)}\|^2 \\
&\leq F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 - \frac{\beta\eta}{4}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\
&\quad + \frac{\beta\eta}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(\bar{x}_t) - \nabla F^{(k)}(x_t^{(k)})\|^2 + \frac{\beta\eta}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(x_t^{(k)}) - m_{t+1}^{(k)}\|^2 \\
&\leq F(\bar{x}_t) - \frac{\beta\eta}{2}\|\nabla F(\bar{x}_t)\|^2 - \frac{\beta\eta}{4}\frac{1}{K}\sum_{k=1}^{K}\|m_{t+1}^{(k)}\|^2 \\
&\quad + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\|\bar{x}_t - x_t^{(k)}\|^2 + \frac{\beta\eta}{K}\sum_{k=1}^{K}\|\nabla F^{(k)}(x_t^{(k)}) - m_{t+1}^{(k)}\|^2 \,,
\end{aligned}
\tag{38}
$$

where the last inequality is due to $\eta \leq \frac{1}{2\beta L_F}$. By introducing the potential function

$$
P_t = \mathbb{E}[F(\bar{x}_t)] + \frac{\beta}{\alpha}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|m_{t+1}^{(k)} - \nabla F^{(k)}(x_t^{(k)})\|^2] + \frac{2\beta C_g^2 L_f^2}{\gamma}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] \,, \tag{39}
$$

one can get

$$
\begin{aligned}
&P_{t+1} - P_t \\
&\leq -\frac{\beta\eta}{2}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] + (\frac{2\eta\beta^3 C_g^4 L_f^2}{\gamma^2} + \frac{2\eta^2\beta^3 C_g^2 L_f^2 C_g^2}{\gamma} + \frac{2\eta\beta^3 L_F^2}{\alpha^2} - \frac{\beta\eta}{4})\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|m_{t+1}^{(k)}\|^2] \\
&\quad + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] + \Big(\beta\eta - \eta\beta\Big)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|m_{t+1}^{(k)} - \nabla F^{(k)}(x_t^{(k)})\|^2] \\
&\quad + \Big(2(1-\gamma\eta)\eta\beta C_g^2 L_f^2 - 2\eta\beta C_g^2 L_f^2\Big)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|u_{t+1}^{(k)} - g^{(k)}(x_t^{(k)})\|^2] \\
&\quad + 2\beta\gamma\eta^2\delta_g^2 C_g^2 L_f^2 + 2\alpha\beta\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\beta\gamma^2\eta^3\delta_g^2 C_g^2 L_f^2 \,.
\end{aligned}
\tag{40}
$$

By setting $\beta \leq \sqrt{\frac{1}{8}\Big/\Big(\frac{C_g^4 L_f^2}{\gamma^2} + \frac{C_g^4 L_f^2}{\gamma} + \frac{L_F^2}{\alpha^2}\Big)}$, one can get

$$
\begin{aligned}
P_{t+1} - P_t \leq &-\frac{\beta\eta}{2}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] + \frac{\beta\eta L_F^2}{K}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] \\
&+ 2\beta\gamma\eta^2 C_g^2 L_f^2\delta_g^2 + 2\alpha\beta\eta^2(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\beta\gamma^2\eta^3 C_g^2 L_f^2\delta_g^2 \,.
\end{aligned}
\tag{41}
$$

By summing t from 0 to $T-1$, one can get

$$\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] \leq \frac{2(P_0-P_T)}{\beta\eta T} + 2\gamma\eta C_g^2 L_f^2\delta_g^2 + 2\alpha\eta(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\gamma^2\eta^2 C_g^2 L_f^2\delta_g^2 \\
&\quad + \frac{2L_F^2}{TK}\sum_{t=0}^{T-1}\sum_{k=1}^{K}\mathbb{E}[\|\bar{x}_t - x_t^{(k)}\|^2] \\
&\leq \frac{2(P_0-P_T)}{\beta\eta T} + 2\gamma\eta C_g^2 L_f^2\delta_g^2 + 2\alpha\eta(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\gamma^2\eta^2 C_g^2 L_f^2\delta_g^2 + 32p^2\beta^2\eta^2 L_F^2(16C_g^2L_f^2\delta_g^2 + C_g^2\sigma_f^2 + 3C_f^2\sigma_g^2) \\
&\leq \frac{2(F(x_0)-F(x_*))}{\beta\eta T} + \frac{\mathbb{E}[\|m_1 - \nabla F(x_0)\|^2]}{\alpha\eta T} + \frac{2C_g^2L_g^2\mathbb{E}[\|u_1 - g(x_0)\|^2]}{\gamma\eta T} \\
&\quad + 2\gamma\eta C_g^2 L_f^2\delta_g^2 + 2\alpha\eta(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\gamma^2\eta^2 C_g^2 L_f^2\delta_g^2 + 32p^2\beta^2\eta^2 L_F^2(16C_g^2L_f^2\delta_g^2 + C_g^2\sigma_f^2 + 3C_f^2\sigma_g^2)\,,
\end{aligned} \tag{42}$$

where the second step holds due to Lemma 4.3, the last step holds due to the definition of the potential function and x_* is the optimal solution.

Additionally, when $t=0$, one can get

$$\begin{aligned}
&\mathbb{E}[\|m_1^{(k)} - \nabla F^{(k)}(x_0)\|^2] \\
&= \mathbb{E}[\|\nabla g^{(k)}(x_0;\xi_0^{(k)})^T\nabla_g f^{(k)}(g^{(k)}(x_0;\xi_0^{(k)});\zeta_0^{(k)}) - \nabla g^{(k)}(x_0)^T\nabla_g f^{(k)}(g^{(k)}(x_0))\|^2] \\
&= \mathbb{E}[\|\nabla g^{(k)}(x_0;\xi_0^{(k)})^T\nabla_g f^{(k)}(g^{(k)}(x_0;\xi_0^{(k)});\zeta_0^{(k)}) - \nabla g^{(k)}(x_0;\xi_0^{(k)})^T\nabla_g f^{(k)}(g^{(k)}(x_0;\xi_0^{(k)})) \\
&\quad + \nabla g^{(k)}(x_0;\xi_0^{(k)})^T\nabla_g f^{(k)}(g^{(k)}(x_0;\xi_0^{(k)})) - \nabla g^{(k)}(x_0;\xi_0^{(k)})^T\nabla_g f^{(k)}(g^{(k)}(x_0)) \\
&\quad + \nabla g(x_0;\xi)^T\nabla_g f(g(x_0)) - \nabla g(x_0)^T\nabla_g f(g(x_0))\|^2] \\
&\leq 3C_g^2\sigma_f^2 + 3C_g^2L_f^2\delta_g^2 + 3C_f^2\sigma_g^2\,.
\end{aligned} \tag{43}$$

where the last step follows from Assumptions 3.2, 3.3. Additionally, one can also get

$$\mathbb{E}[\|u_1^{(k)} - g^{(k)}(x_0)\|^2] = \mathbb{E}[\|g^{(k)}(x_0;\xi_0^{(k)}) - g^{(k)}(x_0)\|^2] = \delta_g^2\,. \tag{44}$$

Then, it is easy to get

$$\begin{aligned}
P_0 &= \mathbb{E}[F(x_0)] + \frac{\beta}{\alpha}\mathbb{E}[\|m_1 - \nabla F(x_0)\|^2] + \frac{2\beta C_g^2L_g^2}{\gamma}\mathbb{E}[\|u_1 - g(x_0)\|^2] \\
&\leq F(x_0) + \frac{\beta(3C_g^2\sigma_f^2 + 3C_g^2L_f^2\delta^2 + 3C_f^2\sigma_g^2)}{\alpha} + \frac{2\beta C_g^2L_g^2\delta^2}{\gamma}.
\end{aligned} \tag{45}$$

By plugging it into Eq. (42), one can get

$$\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\|\nabla F(\bar{x}_t)\|^2] \leq \frac{2(F(x_0)-F(x_*))}{\beta\eta T} + \frac{6(C_g^2\sigma_f^2 + C_g^2L_f^2\delta^2 + C_f^2\sigma_g^2)}{\alpha\eta T} + \frac{4C_g^2L_g^2\delta^2}{\gamma\eta T} \\
&\quad + 2\gamma\eta C_g^2 L_f^2\delta_g^2 + 2\alpha\eta(C_g^2\sigma_f^2 + C_f^2\sigma_g^2) + 2\gamma^2\eta^2 C_g^2 L_f^2\delta_g^2 + 32p^2\beta^2\eta^2 L_F^2(16C_g^2L_f^2\delta_g^2 + C_g^2\sigma_f^2 + 3C_f^2\sigma_g^2)\,.
\end{aligned} \tag{46}$$

□