Projection-Free Variance Reduction Methods for Stochastic Constrained Multi-Level Compositional Optimization

Wei Jiang [1] **Sifan Yang** [1 2] **Wenhao Yang** [1 2] **Yibo Wang** [1 2] **Yuanyu Wan** [3 1] **Lijun Zhang** [1 2]

Abstract

This paper investigates projection-free algorithms for stochastic constrained multi-level optimization. In this context, the objective function is a nested composition of several smooth functions, and the decision set is closed and convex. Existing projection-free algorithms for solving this problem suffer from two limitations: 1) they solely focus on the gradient mapping criterion and fail to match the optimal sample complexities in unconstrained settings; 2) their analysis is exclusively applicable to non-convex functions, without considering convex and strongly convex objectives. To address these issues, we introduce novel projection-free variance reduction algorithms and analyze their complexities under different criteria. For gradient mapping, our complexities improve existing results and match the optimal rates for unconstrained problems. For the widely-used Frank-Wolfe gap criterion, we provide theoretical guarantees that align with those for single-level problems. Additionally, by using a stage-wise adaptation, we further obtain complexities for convex and strongly convex functions. Finally, numerical experiments on different tasks demonstrate the effectiveness of our methods.

1. Introduction

In this paper, we consider projection-free algorithms for stochastic constrained multi-level compositional optimization in the form of

$$\min_{\mathbf{x}\in\mathcal{X}} F(\mathbf{x}) = f_K \circ f_{K-1} \circ \cdots \circ f_1(\mathbf{x}), \tag{1}$$

where $\mathcal{X}$ is a closed convex set. We assume that each function f_i and its gradient can only be accessed through noisy estimations, symbolized as $f_i(\cdot;\xi)$ and $\nabla f_i(\cdot;\xi)$ such that

$$\mathbb{E}_\xi\left[f_i(\cdot;\xi)\right] = f_i(\cdot) \quad \text{and} \quad \mathbb{E}_\xi\left[\nabla f_i(\cdot;\xi)\right] = \nabla f_i(\cdot),$$

where ξ denotes samples drawn from the oracle. Problem (1) finds wide applications in machine learning tasks, such as reinforcement learning (Dann et al., 2014), government planning (Bruno et al., 2016), risk management (Dentcheva et al., 2017), model-agnostic meta-learning (Ji et al., 2020), robust learning (Li et al., 2021), risk-averse portfolio optimization (Shapiro et al., 2021), and graph neural network training (Balasubramanian et al., 2021).

Although stochastic multi-level optimization has been investigated extensively in recent years (Yang et al., 2019; Balasubramanian et al., 2021; Zhang & Xiao, 2021; Chen et al., 2021; Jiang et al., 2022b), current work mainly focuses on unconstrained problems, i.e., $\mathcal{X} = \mathbb{R}^d$. For many practical problems, such as risk-averse portfolio optimization, the decision set is constrained (e.g., the decision variable $\mathbf{x}$ should be in a simplex for portfolio optimization). Traditional constrained optimization typically employs a projection operation to ensure that the solutions are within the decision set. However, projection is usually complicated and time-consuming, and existing literature (Xiao et al., 2022) begin to show interest in developing projection-free algorithms for constrained multi-level problems by replacing projection (a convex optimization problem) with multiple steps of more efficient linear minimization operation.

Projection-free methods typically require two oracles: 1) the Stochastic First-order Oracle (SFO), which takes a point $\mathbf{x}$ and returns the pair $(f(\mathbf{x};\xi), \nabla f(\mathbf{x};\xi))$, where ξ is a sample drawn from the oracle; 2) the Linear Minimization Oracle (LMO), which takes a direction $\mathbf{d}$ and outputs $\arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}, \mathbf{d}\rangle$. To evaluate different projection-free algorithms, the most widely used measures are the number of calls to SFO and LMO required to attain an acceptable solution. For non-convex functions, such a solution $\mathbf{x}$ is usually defined by the Frank-Wolfe gap (Lacoste-Julien, 2016), formalized as

$$\mathcal{F}(\mathbf{x}) := \max_{\hat{\mathbf{x}}\in\mathcal{X}} \langle \hat{\mathbf{x}} - \mathbf{x}, -\nabla F(\mathbf{x})\rangle \leq \epsilon, \tag{2}$$

[1]National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China [2]School of Artificial Intelligence, Nanjing University, Nanjing, China [3]School of Software Technology, Zhejiang University, Ningbo, China. Correspondence to: Lijun Zhang <zhanglj@lamda.nju.edu.cn>.

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

Table 1. Summary of results for projection-free algorithms under three different criteria: the Frank-Wolfe gap (FG), gradient mapping (GM), and optimal gap (OG). Here CVX represents convex functions and SC stands for λ-strongly convex functions. We compare our methods with 1-SFW (Zhang et al., 2019),SPIDER-FW (Yurtsever et al., 2019), NCGS (Qu et al., 2018), SGD+ICG (Balasubramanian & Ghadimi, 2018), LiNASA+ICG (Xiao et al., 2022), and SCGS (Lan & Zhou, 2016).

Method	Criterion	Assumptions	Level	Batch size	SFO	LMO
1-SFW	FG	Smooth	1	1	$\mathcal{O}\left(\epsilon^{-3}\right)$	$\mathcal{O}\left(\epsilon^{-3}\right)$
SPIDER-FW	FG	Smooth	1	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-3}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
Theorem 1	FG	Smooth	K	1	$\mathcal{O}\left(\epsilon^{-3}\right)$	$\mathcal{O}\left(\epsilon^{-3}\right)$
Theorem 2	FG	Smooth	K	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-3}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
NCGS	GM	Smooth	1	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
SGD+ICG	GM	Smooth	1	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
LiNASA+ICG	GM	Smooth	K	1	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-3}\right)$
Theorem 3	GM	Smooth	K	1	$\mathcal{O}\left(\epsilon^{-1.5}\right)$	$\mathcal{O}\left(\epsilon^{-2.5}\right)$
Theorem 4	GM	Smooth	K	$\mathcal{O}\left(\epsilon^{-0.5}\right)$	$\mathcal{O}\left(\epsilon^{-1.5}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
1-SFW	OG	Smooth+CVX	1	1	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
SPIDER-FW	OG	Smooth+CVX	1	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-1}\right)$
Theorem 5	OG	Smooth+CVX	K	1	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
Theorem 6	OG	Smooth+CVX	K	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$	$\mathcal{O}\left(\epsilon^{-1}\right)$
SCGS	OG	Smooth+SC	1	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\lambda^{-1}\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-1}\right)$
Theorem 7	OG	Smooth+SC	K	1	$\mathcal{O}\left(\lambda^{-1}\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-2}\right)$
Theorem 8	OG	Smooth+SC	K	$\mathcal{O}\left(\epsilon^{-1}\right)$	$\mathcal{O}\left(\lambda^{-1}\epsilon^{-1}\right)$	$\mathcal{O}\left(\epsilon^{-1}\right)$

where ϵ is a small value. More recently, the gradient mapping criterion (Qu et al., 2018) has been introduced, which is denoted as

$$\mathcal{G}(\mathbf{x}) := \left\| \beta \left(\mathbf{x} - \Pi_{\mathcal{X}} \left(\mathbf{x} - \frac{1}{\beta} \nabla F(\mathbf{x}) \right) \right) \right\|^2 \leq \epsilon, \quad (3)$$

where $\Pi_{\mathcal{X}}$ denote the projection onto the domain $\mathcal{X}$ and parameter β is a positive constant. Notably, if $\mathcal{X} = \mathbb{R}^d$, this gradient mapping criterion simplifies to the stationary point, i.e., $\mathbb{E}\left[\|\nabla F(\mathbf{x})\|^2\right] \leq \epsilon$, which is the standard metric for stochastic unconstrained problems. When the objective function is convex or strongly convex, the optimal gap criterion is employed instead, expressed as

$$F(\mathbf{x}) - \min_{\hat{\mathbf{x}} \in \mathcal{X}} F(\hat{\mathbf{x}}) \leq \epsilon, \quad (4)$$

which measures the difference between the objective and the optimal value.

The current method for stochastic projection-free multi-level compositional optimization, named as LiNASA+ICG (Xiao et al., 2022), integrates the linearized NASA (Ghadimi et al., 2020) algorithm with inexact conditional gradient (Balasubramanian & Ghadimi, 2018), and provides theoretical guarantees under the gradient mapping criterion. This method can identify an acceptable point with $\mathcal{O}(\epsilon^{-2})$ calls to SFO and $\mathcal{O}(\epsilon^{-3})$ calls to LMO. However, it still suffers from several drawbacks. Firstly, its SFO complexity does not match the existing optimal rate of $\mathcal{O}(\epsilon^{-1.5})$ for stochastic unconstrained problems. Secondly, its analysis solely concentrates on gradient mapping, and results for the more widely used Frank-Wolfe gap criterion are not provided. Finally, this method is restricted to non-convex objective functions, and it is unclear how to improve the rate for convex and strongly convex objectives.

To address these issues, we propose new algorithms that utilize the variance reduction estimator STORM (Cutkosky & Orabona, 2019) to obtain more accurate evaluations of both the inner function values and the overall gradient. By integrating these estimators with a specifically designed Frank-Wolfe algorithm (Jaggi, 2013), we improve the rate for gradient mapping and are able to analyze the Frank-Wolfe gap. Besides, by employing a large batch size, we can reduce the iteration numbers and thus improve the LMO complexities while maintaining the same SFO rates. Moreover, we develop a stage-wise algorithm with a warm-start technique and provide theoretical guarantees for both convex and strongly convex functions. Compared with previous methods, this paper makes the following contributions:

1. We establish the first theoretical guarantees for the Frank-Wolfe gap (Theorem 1, 2) under the multi-level setting. The rates we obtained match the results for single-level projection-free methods (Zhang et al., 2019; Yurtsever et al., 2019).
2. For gradient mapping (Theorem 3, 4), our approach achieves an improved SFO complexity of $\mathcal{O}(\epsilon^{-1.5})$ and LMO complexity of $\mathcal{O}(\epsilon^{-2.5})$. The SFO complexity matches the low bound (Arjevani et al., 2019), and the

LMO complexity can be further reduced to $\mathcal{O}(\epsilon^{-2})$ by using large batch sizes.

3. We explore the complexities for convex (Theorem 5, 6) and strongly convex functions (Theorem 7, 8), and derive the optimal SFO rates for these problems, which have not been studied in previous projection-free multi-level literature.

We compare our theoretical results with existing methods in Table 1, and validate the effectiveness of our method through numerical experiments in Section 4.

2. Related Work

This section briefly reviews related work on stochastic multi-level compositional optimization and stochastic projection-free algorithms.

2.1. Stochastic Multi-Level Compositional Optimization

Stochastic Compositional Optimization has been explored extensively in the literature, and most research focuses on the two-level settings (Wang et al., 2017a;b; Ghadimi et al., 2020; Zhang & Xiao, 2019; Chen et al., 2021; Qi et al., 2021; Jiang et al., 2022a; 2023; Yu et al., 2024). The problem of multi-level compositional optimization was first investigated by Yang et al. (2019). Inspired by multi-timescale stochastic approximation (Wang et al., 2017a), they introduced the multi-level stochastic gradient method, which achieves a sample complexity of $\mathcal{O}\left(1/\epsilon^{(7+K)/2}\right)$ for K-level problems. When the function is strongly convex, this complexity can be further improved to $\mathcal{O}\left(1/\epsilon^{(3+K)/4}\right)$. Subsequently, motivated by the NASA algorithm (Ghadimi et al., 2020), Balasubramanian et al. (2021) proposed using a linearized averaging stochastic estimator to track the function value, attaining a sample complexity of $\mathcal{O}\left(1/\epsilon^4\right)$ for non-convex objectives. This rate was also obtained in a concurrent work (Chen et al., 2021) by employing variance reduction techniques to evaluate the function value.

Later, Zhang & Xiao (2021) employed nested variance reduction to approximate gradients, improving the sample complexity to the optimal $\mathcal{O}\left(1/\epsilon^3\right)$. However, this approach requires a large and increasing batch size on the order of $\mathcal{O}\left(1/\epsilon\right)$. To address this issue, Jiang et al. (2022b) developed a method called SMVR, which achieves the same optimal rate but only requires a constant batch size. SMVR also attains an improved rate of $\mathcal{O}\left(1/\epsilon^2\right)$ for convex functions and $\mathcal{O}\left(1/\left(\lambda\epsilon\right)\right)$ for λ-strongly convex objectives. More recently, Gao (2023) further introduced the decentralized stochastic multi-level optimization algorithm, which achieves the level-independent convergence rate under the decentralized setting. Despite these advancements, these algorithms are only applicable to unconstrained problems.

2.2. Stochastic Projection-Free Algorithms

The most well-known projection-free method, Frank-Wolfe algorithm (Frank & Wolfe, 1956), was originally designed for smooth convex optimization with polyhedral domains and has been extended to any convex compact set by Jaggi (2013). In the stochastic setting, Hazan & Kale (2012) first developed a projection-free method for online smooth convex optimization. Later, Hazan & Luo (2016) applied variance reduction techniques to the stochastic Frank-Wolfe algorithm. Inspired by the accelerated gradient method (Nesterov, 1983), Lan & Zhou (2016) proposed the stochastic conditional gradient sliding (SCGS) method, which offered an SFO complexity of $\mathcal{O}\left(\lambda^{-1}\epsilon^{-1}\right)$ and an LMO complexity of $\mathcal{O}\left(\epsilon^{-1}\right)$ for smooth λ-strongly convex optimization. Besides that, projection-free methods are also widely investigated in online convex optimization in recent years (Hazan & Minasyan, 2020; Wan et al., 2020; 2022; Mhammedi, 2022; Garber & Kretzu, 2023; Wang et al., 2024).

For non-convex objectives, Reddi et al. (2016) introduced the SVFW method, achieving an SFO complexity of $\mathcal{O}(\epsilon^{-10/3})$ and an LMO complexity of $\mathcal{O}(\epsilon^{-2})$ under the Frank-Wolfe gap. Motivated by the variance reduction technique SPIDER (Fang et al., 2018), Yurtsever et al. (2019) developed the SPIDER-FW algorithm, improving the SFO complexity to $\mathcal{O}(\epsilon^{-3})$ by using a large batch size of $\mathcal{O}(\epsilon^{-1})$. To avoid relying on large batches, Zhang et al. (2019) proposed the one-sample stochastic Frank-Wolfe algorithm (1-SFW), which attains the same SFO complexity and obtains an LMO complexity of $\mathcal{O}(\epsilon^{-3})$. Rather than focusing on the Frank-Wolfe gap criterion, Qu et al. (2018) and Balasubramanian & Ghadimi (2018) explored the gradient mapping criterion, and attained $\mathcal{O}(\epsilon^{-2})$ complexities for both the SFO and LMO at the cost of using a large batch of $\mathcal{O}(\epsilon^{-1})$ in each iteration.

In the context of stochastic multi-level optimization, Xiao et al. (2022) recently proposed a projection-free conditional gradient-type algorithm, which combines the linearized NASA algorithm with the inexact conditional gradient technique (Balasubramanian & Ghadimi, 2018). This method achieves an SFO complexity of $\mathcal{O}\left(1/\epsilon^2\right)$ and an LMO complexity of $\mathcal{O}\left(1/\epsilon^3\right)$ in terms of the gradient mapping criterion. However, its SFO complexity does not match the complexity of $\mathcal{O}\left(1/\epsilon^{1.5}\right)$ achieved by variance reduction methods for unconstrained multi-level problems[1]. Furthermore, they only consider non-convex functions, and solely focus on the gradient mapping criterion, which are the main drawbacks we aim to address in this paper.

[1]Gradient mapping reduces to $\mathbb{E}\left[\|\nabla F(\mathbf{x})\|^2\right] \leq \epsilon$ for unconstrained problems, and existing methods (Zhang & Xiao, 2021; Jiang et al., 2022b) ensure $\mathbb{E}\left[\|\nabla F(\mathbf{x})\|\right] \leq \epsilon$ with a complexity of $\mathcal{O}(1/\epsilon^3)$, implying a rate of $\mathcal{O}(1/\epsilon^{1.5})$ for gradient mapping.

3. The Proposed Methods

In this section, we first introduce the assumptions used in this paper. Then, we present the proposed algorithms, along with their corresponding theoretical guarantees for three different criteria: Frank-Wolfe gap, gradient mapping and optimal gap.

3.1. Assumptions

We adopt the following assumptions throughout the paper, which are commonly used in studies of stochastic compositional optimization (Yuan et al., 2019; Zhang & Xiao, 2019; 2021; Jiang et al., 2022b) and stochastic projection-free analysis (Qu et al., 2018; Yurtsever et al., 2019; Zhang et al., 2019).

Assumption 1. *(Constrained set) The decision set $\mathcal{X}$ is closed and convex with a bounded domain such that $\max_{\mathbf{x},\mathbf{y}\in\mathcal{X}} \|\mathbf{x}-\mathbf{y}\| \leq D$.*

Assumption 2. *(Smoothness and Lipschitz continuity) All functions $f_1,\ldots,f_K$ are L_f-Lipschitz continuous, and their Jacobians $\nabla f_1,\ldots,\nabla f_K$ are L_J-Lipschitz continuous.*

Assumption 3. *(Bounded variance) For $1\leq i\leq K$, we assume that:*

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

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

Assumption 4. *(Average smoothness) For $1\leq i\leq K$, we assume that:*

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

Assumption 5. *We suppose that $F(\mathbf{x}_1)-F_\star\leq\Delta_F$ for the initial solution $\mathbf{x}_1$, where $F_\star=\min_{\mathbf{x}\in\mathcal{X}}F(\mathbf{x})$.*

3.2. Results for Frank-Wolfe Gap

First, we examine the sample complexity under the criterion of Frank-Wolfe gap. The primary procedure of our algorithm involves estimating the gradient of the objective function and then employing the Frank-Wolfe method to replace the projection operation. Note that the gradient of the multi-level function exhibits a nested structure, and the estimation error would accumulate as the level becomes deeper. To address this issue, we resort to the variance reduction technique STORM (Cutkosky & Orabona, 2019) to estimate both the inner function value and the overall gradient. Specifically, we draw a batch of samples $\{\xi_t^{i,1},\cdots,\xi_t^{i,B_1}\}$ with the batch size B_1 for each level i at time step t, and then employ a variance reduction estimator $\mathbf{u}_t^i$ to track each inner function value $f_i(\cdot)$ as:

$$\begin{aligned}
\mathbf{u}_t^i = (1-\alpha)\mathbf{u}_{t-1}^i &+ \frac{1}{B_1}\sum_{j=1}^{B_1} f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\\
&-(1-\alpha)\frac{1}{B_1}\sum_{j=1}^{B_1} f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}).
\end{aligned}\tag{5}$$

This evaluation ensures that the estimation error is reduced over time. Then, we employ a similar variance-reduced estimator $\mathbf{v}_t$ to evaluate the overall gradient of the objective function $\nabla F(\mathbf{x}_t)$ as:

$$\begin{aligned}
\mathbf{v}_t = (1-\alpha)\mathbf{v}_{t-1} &+ \frac{1}{B_1}\sum_{j=1}^{B_1}\left[\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right]\\
&-(1-\alpha)\frac{1}{B_1}\sum_{j=1}^{B_1}\left[\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right].
\end{aligned}\tag{6}$$

After obtaining the gradient estimation, we follow the framework of the Frank-Wolfe algorithm (Jaggi, 2013), but use the estimator $\mathbf{v}_t$ to replace the gradient required in the original algorithm as follows:

$$\begin{aligned}
\mathbf{z}_t &= \arg\min_{\mathbf{x}\in\mathcal{X}}\langle\mathbf{x},\mathbf{v}_t\rangle,\\
\mathbf{x}_{t+1} &= \mathbf{x}_t+\eta(\mathbf{z}_t-\mathbf{x}_t).
\end{aligned}$$

In this way, we develop our Projection-free Multi-level Variance Reduction (PMVR) method for stochastic multi-level problems. The complete algorithm is presented in Algorithm 1. Note that in the first iteration (when $t=1$), we can simply set estimator $\mathbf{u}_1^i=\frac{1}{B_0}\sum_{j=1}^{B_0}f_i(\mathbf{u}_1^{i-1};\xi_1^{i,j})$ and $\mathbf{v}_1=\frac{1}{B_0}\sum_{j=1}^{B_0}\left[\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1};\xi_1^{i,j})\right]$, where B_0 is the batch size for the first iteration.

Comparison with the SMVR method: Although the SMVR algorithm (Jiang et al., 2022b) for unconstrained problems also utilizes the STORM estimator to assess inner function values and gradients at each level, our PMVR method differs from SMVR in the following aspects: i) SMVR first estimates the gradient at each level using STORM and then computes the overall gradient through multiplication, requiring an additional gradient clipping for each level to ensure the overall gradient does not explode. This operation demands knowledge of the gradient upper bound of each level, which is impractical in real-world applications. In contrast, our PMVR method directly applies variance reduction to the overall gradient, eliminating the need for such gradient clipping. This technique is also mentioned in a concurrent work (Gao, 2023); ii) Our method employs a constant learning rate η and momentum parameter α,

Algorithm 1 PMVR Algorithm

Input: parameters T, η, α, initial points $(\mathbf{x}_1, \mathbf{u}_1, \mathbf{v}_1)$
for time step $t = 1$ **to** T **do**
  Set $\mathbf{u}_t^0 = \mathbf{x}_t$
  for level $i = 1$ **to** K **do**
    Draw a batch of samples $\left\{\xi_t^{i,1}, \ldots, \xi_t^{i,B_1}\right\}$
    Compute the estimator $\mathbf{u}_t^i$ according to (5)
  end for
  Compute the gradient estimator $\mathbf{v}_t$ according to (6)
  Compute $\mathbf{z}_t = \arg\min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{x}, \mathbf{v}_t \rangle$
  Update the weight: $\mathbf{x}_{t+1} = \mathbf{x}_t + \eta(\mathbf{z}_t - \mathbf{x}_t)$
end for
Choose τ uniformly at random from the set $\{1, \ldots, T\}$
Return $(\mathbf{x}_\tau, \mathbf{u}_\tau, \mathbf{v}_\tau)$

which are easy to implement and fine-tune, whereas SMVR requires these parameters to decrease gradually, which is more complicated; iii) Our PMVR is a projection-free algorithm specifically designed for constrained problems, with theoretical guarantees focused on the Frank-Wolfe gap and gradient mapping. In contrast, SMVR aims to find stationary points for unconstrained objectives.

Next, we present the sample complexities of Algorithm 1 with respect to the Frank-Wolfe gap $\mathcal{F}(\cdot)$ defined in equation (2). Note that by using a large batch size B_1, we are able to decrease the iteration numbers and thus reduce the LMO complexity. As a result, we provide a constant batch version and a large batch version for our guarantees.

Theorem 1. *By setting $B_1 = \mathcal{O}(1)$, $\eta = \mathcal{O}(\epsilon^2)$, and $\alpha = \mathcal{O}(\epsilon^2)$, our PMVR algorithm guarantees that $\mathbb{E}\left[\mathcal{F}(\mathbf{x}_\tau)\right] \leq \epsilon$ within $T = \mathcal{O}(\epsilon^{-3})$ iterations.*

Remark: The above theorem indicates that both the SFO and LMO complexities are of the order $\mathcal{O}(\epsilon^{-3})$, consistent with results for projection-free single-level problems using a constant batch size (Zhang et al., 2019).

Theorem 2. *(Large Batch) By setting $B_1 = \mathcal{O}(\epsilon^{-1})$, $\eta = \mathcal{O}(\epsilon)$, and $\alpha = \mathcal{O}(\epsilon)$, our PMVR algorithm guarantees that $\mathbb{E}\left[\mathcal{F}(\mathbf{x}_\tau)\right] \leq \epsilon$ within $T = \mathcal{O}(\epsilon^{-2})$ iterations.*

Remark: By employing a large batch size of $\mathcal{O}(\epsilon^{-1})$, our method obtains an SFO complexity of $\mathcal{O}(\epsilon^{-3})$ and an LMO complexity of $\mathcal{O}(\epsilon^{-2})$. These results align with those achieved by existing projection-free methods for single-level objectives (Yurtsever et al., 2019).

3.3. Results for Gradient Mapping

Then, we investigate the complexities under the criterion of gradient mapping $\mathcal{G}(\cdot)$ defined in equation (3). To deal with the gradient mapping, our PMVR algorithm only requires minimal modifications to fit this criterion. Based on

Algorithm 2 PMVR-v2

Replace Step 9 of Algorithm 1 with the following
Initialize $\mathbf{w}_1 = \mathbf{x}_t$
for time step $n = 1$ **to** N **do**
  Compute $\mathbf{s} = \arg\min_{\hat{\mathbf{s}}\in\mathcal{X}} \langle \mathbf{v}_t + \beta(\mathbf{w}_n - \mathbf{x}_t), \hat{\mathbf{s}} \rangle$
  Set $\mathbf{w}_{n+1} = (1-\gamma_t)\mathbf{w}_n + \gamma_t \mathbf{s}$
end for
Set $\mathbf{z}_t = \mathbf{w}_{N+1}$

Proposition 2 of Xiao et al. (2022), gradient mapping can be decomposed into two components:

$$\mathcal{G}(\mathbf{x}_t) \leq -4\beta \min_{\mathbf{w}\in\mathcal{X}} g(\mathbf{w}, \mathbf{x}_t, \mathbf{v}_t) + 2\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2,$$

where

$$g(\mathbf{w}, \mathbf{x}_t, \mathbf{v}_t) = \langle \mathbf{v}_t, \mathbf{w} - \mathbf{x}_t \rangle + \frac{\beta}{2}\|\mathbf{w} - \mathbf{x}_t\|^2.$$

Since our PMVR method has already employed variance-reduced techniques to ensure that the gradient estimation error $\mathbb{E}\left[\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2\right]$ decreases over time, we can reuse this part and focus on bounding the $-\min_{\mathbf{w}\in\mathcal{X}} g(\mathbf{w}, \mathbf{x}_t, \mathbf{v}_t)$ term. To address this constrained quadratic minimization problem, we design a sub-algorithm, modified from the Frank-Wolfe method (Jaggi, 2013), which runs for several loops. In each loop n, we update as follows:

$$\mathbf{s} = \arg\min_{\hat{\mathbf{s}}\in\mathcal{X}} \langle \mathbf{v}_t + \beta(\mathbf{w}_n - \mathbf{x}_t), \hat{\mathbf{s}} \rangle,$$
$$\mathbf{w}_{n+1} = (1-\gamma_t)\mathbf{w}_n + \gamma_t \mathbf{s}.$$

Note that $\mathbf{v}_t + \beta(\mathbf{w}_n - \mathbf{x}_t)$ is the gradient of $g(\mathbf{w}_n, \mathbf{x}_t, \mathbf{v}_t)$ with respect to parameter $\mathbf{w}_n$. Overall, we only need to replace Step 9 in Algorithm 1 with a sub-algorithm presented in Algorithm 2, and we can obtain the guarantees for gradient mapping below.

Theorem 3. *By setting $B_1 = \mathcal{O}(1)$, $N = \mathcal{O}(\epsilon^{-1})$, $\eta = \mathcal{O}(\sqrt{\epsilon})$, and $\alpha = \mathcal{O}(\epsilon)$, our PMVR-v2 algorithm guarantees $\mathbb{E}\left[\mathcal{G}(\mathbf{x}_\tau)\right] \leq \epsilon$ in $T = \mathcal{O}(\epsilon^{-1.5})$ iterations.*

Remark: When using a constant batch size, our algorithm results in $\mathcal{O}(\epsilon^{-1.5})$ SFO complexity and $\mathcal{O}(\epsilon^{-2.5})$ LMO complexity, which are both better than the previous algorithm LiNASA+ICG (Xiao et al., 2022), that achieves $\mathcal{O}(\epsilon^{-2})$ SFO complexity and $\mathcal{O}(\epsilon^{-3})$ LMO complexity.

Next, we can improve the LMO complexity by using a large batch size.

Theorem 4. *(Large Batch) By setting $B_1 = \mathcal{O}(\epsilon^{-0.5})$, $N = \mathcal{O}(\epsilon^{-1})$, $\eta = \mathcal{O}(1)$, and $\alpha = \mathcal{O}(\sqrt{\epsilon})$, our PMVR-v2 ensures $\mathbb{E}\left[\mathcal{G}(\mathbf{x}_\tau)\right] \leq \epsilon$ in $T = \mathcal{O}(\epsilon^{-1})$ iterations.*

Remark: The above theorem indicates that PMVR-v2, with a batch size of $\mathcal{O}(\epsilon^{-0.5})$, achieves an SFO complexity

Algorithm 3 Stage-wise PMVR

Input: initial points $(\mathbf{x}_0, \mathbf{u}_0, \mathbf{v}_0)$
for stage $s = 1$ **to** S **do**
  $(\mathbf{x}_s, \mathbf{u}_s, \mathbf{v}_s)$ = Algorithm 1 with parameters T_s, η_s, α_s and initial points $(\mathbf{x}_{s-1}, \mathbf{u}_{s-1}, \mathbf{v}_{s-1})$
end for
Return $\mathbf{x}_S$

of $\mathcal{O}(\epsilon^{-1.5})$ and an LMO complexity of $\mathcal{O}(\epsilon^{-2})$. These rates are superior to methods for single-level objectives, i.e., NCGS (Qu et al., 2018) and SGD+ICG (Balasubramanian & Ghadimi, 2018), which require a larger batch size of $\mathcal{O}(\epsilon^{-1})$ and demand a worse SFO complexity of $\mathcal{O}\left(\epsilon^{-2}\right)$. Notably, our SFO complexity of $\mathcal{O}(\epsilon^{-1.5})$ also matches the lower bound for stochastic unconstrained optimization (Arjevani et al., 2019).

3.4. Results for Optimal Gap

In addition to the analyses for general non-convex functions in previous subsections, we further investigate the case for convex and strongly convex functions, defined as follows.

Definition 1. *A function $F : \mathcal{X} \mapsto \mathbb{R}$ is convex if*

$$F(\mathbf{y}) \geq F(\mathbf{x}) + \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle, \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}.$$

Definition 2. *A function $F : \mathcal{X} \mapsto \mathbb{R}$ is λ-strongly convex if $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$*

$$F(\mathbf{y}) \geq F(\mathbf{x}) + \langle \nabla F(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\lambda}{2}\|\mathbf{y} - \mathbf{x}\|^2.$$

When the objective function is convex or strongly convex, a local optimal point becomes a global optimal point. Consequently, a natural criterion to consider is the optimal gap defined in equation (4).

Convex functions: First, we investigate the case for convex functions. To obtain theoretical guarantees, we design a stage-wise algorithm with the warm-start technique building on Algorithm 1. Specifically, we divide the entire process into S stages, and for each stage s, we run Algorithm 1 with a new set of parameters T_s, η_s, α_s, and use the output of the previous stage $\{\mathbf{x}_{s-1}, \mathbf{u}_{s-1}, \mathbf{v}_{s-1}\}$ as the initial points. The complete method is shown in Algorithm 3, referred to as Stage-wise PMVR.

By decreasing α_s and η_s, and increasing T_s at each stage, we can ensure that the optimal gap is halved after each stage, such that $\mathbb{E}\left[F(\mathbf{x}_s) - F_\star\right] \leq \frac{1}{2}\mathbb{E}\left[F(\mathbf{x}_{s-1}) - F_\star\right]$. Denoting $S = \mathcal{O}(\log(\frac{\epsilon_1}{\epsilon}))$ and $\epsilon_s = \epsilon_1/2^s$, where ϵ_1 is a positive constant, we can obtain the guarantees for optimal gap in the following theorem.

Theorem 5. *Setting $B_1 = \mathcal{O}(1)$, $\eta_s = \mathcal{O}(\epsilon_s^2)$, $\alpha_s = \mathcal{O}(\epsilon_s^2)$, and $T_s = \mathcal{O}(\epsilon_s^{-2})$, we ensure $\mathbb{E}\left[F(\mathbf{x}_S)\right] - \min_{\hat{\mathbf{x}}} F(\hat{\mathbf{x}}) \leq \epsilon$ in $\mathcal{O}(\epsilon^{-2})$ iterations.*

Algorithm 4 Stage-wise PMVR-v2

Replace Step 9 of Algorithm 1 with the following
Initialize $\mathbf{w}_1 = \mathbf{x}_t$
for time step $n = 1$ **to** N **do**
  Compute $\mathbf{s} = \arg\min_{\hat{\mathbf{s}} \in \mathcal{X}} \left\langle \mathbf{v}_t + \frac{\lambda}{2}\left(\mathbf{w}_n - \mathbf{x}_t\right), \hat{\mathbf{s}} \right\rangle$
  Set $\mathbf{w}_{n+1} = (1 - \gamma_t)\,\mathbf{w}_n + \gamma_t \mathbf{s}$
end for
Set $\mathbf{z}_t = \mathbf{w}_{N+1}$

Remark: When using a constant batch size, we obtain $\mathcal{O}(\epsilon^{-2})$ complexity for both SFO and LMO, matching the results for single-level problems (Zhang et al., 2019). Also note that the SFO complexity of $\mathcal{O}(\epsilon^{-2})$ is already optimal for convex objective functions (Agarwal et al., 2012).

Theorem 6. *(Large Batch) By setting $B_1 = \mathcal{O}(\epsilon_s^{-1})$, $\eta_s = \mathcal{O}(\epsilon_s)$, $\alpha_s = \mathcal{O}(\epsilon_s)$, and $T_s = \mathcal{O}(\epsilon_s^{-1})$, we can ensure that $\mathbb{E}\left[F(\mathbf{x}_S)\right] - \min_{\hat{\mathbf{x}}} F(\hat{\mathbf{x}}) \leq \epsilon$ in $\mathcal{O}(\epsilon^{-1})$ iterations.*

Remark: The above theorem indicates that our algorithm achieves an SFO complexity of $\mathcal{O}(\epsilon^{-2})$ and an LMO complexity of $\mathcal{O}(\epsilon^{-1})$ with a large batch size of $\mathcal{O}(\epsilon^{-1})$, aligning with the rates of methods for single-level settings, such as Spider-FW (Yurtsever et al., 2019).

Strongly convex functions: Compared with convex functions, we have an additional term $\frac{\lambda}{2}\|\mathbf{y} - \mathbf{x}\|^2$ for strongly convex objectives according to the Definition 2. So, instead of applying $\arg\min_{\mathbf{x} \in \mathcal{X}} \langle \mathbf{x}, \mathbf{v}_t \rangle$ in Step 9 of Algorithm 1, we aim to

$$\min_{\mathbf{w} \in \mathcal{X}} g(\mathbf{w}, \mathbf{x}_t, \mathbf{v}_t) \coloneqq \left\{ \langle \mathbf{w}, \mathbf{v}_t \rangle + \frac{\lambda}{4}\|\mathbf{w} - \mathbf{x}_t\|^2 \right\}$$

via LMO in this case. To this end, we design a sub-algorithm that runs the Frank-Wolfe method (Jaggi, 2013) for several loops. In each loop n, we update as:

$$\begin{aligned} \mathbf{s} &= \arg\min_{\hat{\mathbf{s}} \in \mathcal{X}} \left\langle \mathbf{v}_t + \frac{\lambda}{2}\left(\mathbf{w}_n - \mathbf{x}_t\right), \hat{\mathbf{s}} \right\rangle, \\ \mathbf{w}_{n+1} &= (1 - \gamma_t)\,\mathbf{w}_n + \gamma_t \mathbf{s}, \end{aligned} \tag{7}$$

where $\mathbf{v}_t + \frac{\lambda}{2}\left(\mathbf{w}_n - \mathbf{x}_t\right)$ is the gradient of $g(\mathbf{w}_n, \mathbf{x}_t, \mathbf{v}_t)$ with respect to parameter $\mathbf{w}_n$. Overall, we retain the stage-wise design of the original Algorithm 3, but replace step 9 in its basic Algorithm 1 with the newly developed Algorithm 4. This modification allows us to establish the optimal gap guarantees as detailed below. Here, we denote $\epsilon_s = \epsilon_1/2^s$, where ϵ_1 is a positive constant.

Theorem 7. *Setting $B_1 = \mathcal{O}(1)$, $N = \mathcal{O}(\lambda\epsilon^{-1})$, $\eta_s = \mathcal{O}(\lambda\epsilon_s)$, $\alpha_s = \mathcal{O}(\lambda\epsilon_s)$, and $T_s = \mathcal{O}(\lambda^{-1}\epsilon_s^{-1})$, we ensure $\mathbb{E}\left[F(\mathbf{x}_S)\right] - \min_{\hat{\mathbf{x}}} F(\hat{\mathbf{x}}) \leq \epsilon$ in $S = \mathcal{O}(\log(\frac{\epsilon_1}{\epsilon}))$ stages.*

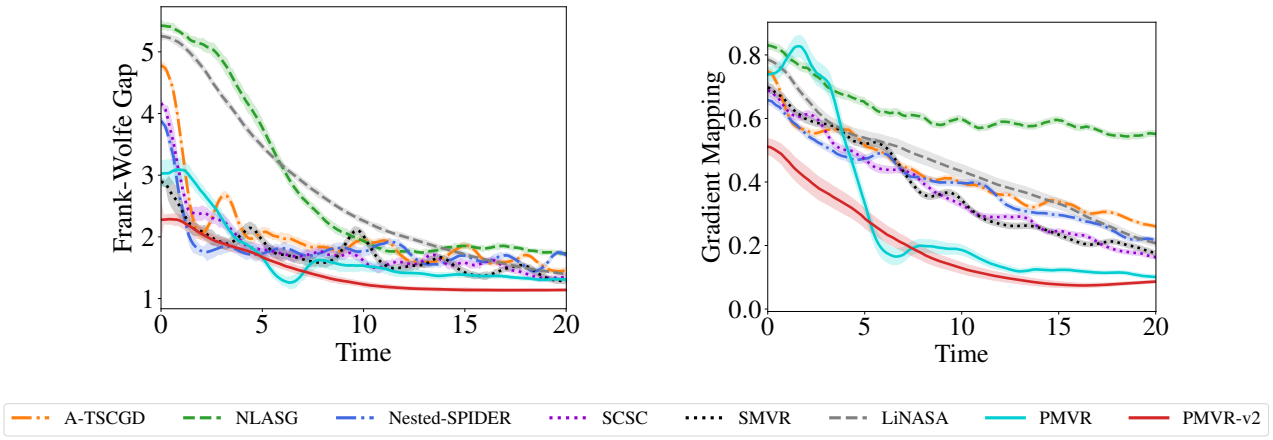


Figure 1. Results for matrix optimization with low-rank constraints.

Remark: When employing constant batch sizes, we can achieve an SFO complexity of $\mathcal{O}(\lambda^{-1}\epsilon^{-1})$ and an LMO complexity of $\mathcal{O}(\epsilon^{-2})$. Notably, the $\mathcal{O}(\lambda^{-1}\epsilon^{-1})$ SFO complexity we obtained is already optimal for stochastic unconstrained strongly convex problems (Agarwal et al., 2012) and is thus unimprovable.

Theorem 8. *(Large Batch) By setting* $B_1 = \mathcal{O}(\epsilon_s^{-1})$, $N = \mathcal{O}(\lambda\epsilon^{-1})$, $\eta_s = \mathcal{O}(\lambda)$, $\alpha_s = \mathcal{O}(\lambda)$, *and* $T_s = \mathcal{O}(\lambda^{-1})$, *we can guarantee that* $\mathbb{E}\left[F(\mathbf{x}_S)\right] - \min_{\hat{\mathbf{x}}} F(\hat{\mathbf{x}}) \leq \epsilon$ *within* $S = \mathcal{O}(\log(\frac{\epsilon_1}{\epsilon}))$ *stages.*

Remark: By using a large batch size, the SFO complexity remains on the same order, and the LMO complexity can be further improved to $\mathcal{O}(\epsilon^{-1})$, matching the existing results for strongly convex functions in the single-level setting (Lan & Zhou, 2016).

4. Experiments

In this section, we evaluate the effectiveness of our proposed methods through numerical experiments on three different problems. We compare our methods with existing stochastic multi-level algorithms, including A-TSCGD (Yang et al., 2019), NLASG (Balasubramanian et al., 2021), Nested-SPIDER (Zhang & Xiao, 2021), SCSC (Chen et al., 2021), and SMVR (Jiang et al., 2022b). We also compare with the previous projection-free multi-level algorithm LiNASA+ICG (Xiao et al., 2022). For our algorithm, we select the momentum parameter α from the set $\{0.01, 0.03, 0.05, 0.1, 0.3\}$ and search the parameter N for PMVR-v2 from the range $\{10, 50, 100\}$. For the other methods, we adopt the hyper-parameters recommended in their original papers or perform a grid search to select the best ones. The learning rate is fine-tuned within the range of $\{0.001, 0.005, 0.01, 0.05, 0.1\}$. All experiments are conducted on a personal laptop.

4.1. Matrix Optimization with Low-Rank Constraints

Following the previous literature on projection-free multi-level optimization (Xiao et al., 2022), we also conduct experiments on matrix optimization with low-rank constraints. Specifically, we consider the matrix-valued single-index model (Yang et al., 2017) with a low-rank constraint, expressed as:

$$y = |\langle A, B^\star\rangle_F|^2 + \epsilon, \quad \text{rank}(B^\star) \leq s,$$

where $A, B \in \mathbb{R}^{m\times n}$, $\epsilon \sim \mathcal{N}\left(0, \sigma^2\right)$, $\langle\cdot,\cdot\rangle_F$ denotes the Frobenius inner product, and s is a positive integer smaller than both m and n. To recover a low-rank matrix $B^\star$ given A and y, we can optimize the mean squared loss function with a nuclear norm constraint. The objective function can be formulated as:

$$\min_B F(B) = \mathbb{E}\left[\left(y - |\langle A, B\rangle_F|^2\right)^2\right]$$
$$\text{s.t.} \quad \|B\|_\star \leq s.$$

Note that in this case, the projection operation onto the nuclear norm ball requires a full singular value decomposition (SVD), while the linear optimization used by Frank-Wolfe update only requires computing the singular vector pair corresponding to the largest singular value, which is much cheaper (Jaggi, 2013). In line with the setup in Xiao et al. (2022), we define the matrix $B^\star = vv^\top / \left\|vv^\top\right\|_\star$, and the matrix A is generated by $A = I + E$, with $E_{i,j} \overset{i.i.d.}{\sim} \mathcal{N}(0, 0.3)$.

In Figure 1, we plot the value of Frank-Wolfe gap, as well as gradient mapping, against the running time of each algorithm. All the curves are averaged over 50 runs. As can be seen, our PMVR and PMVR-v2 methods demonstrate a more rapid decrease in both criteria compared to other approaches, especially for the gradient mapping criterion, demonstrating the superiority of our proposed methods.

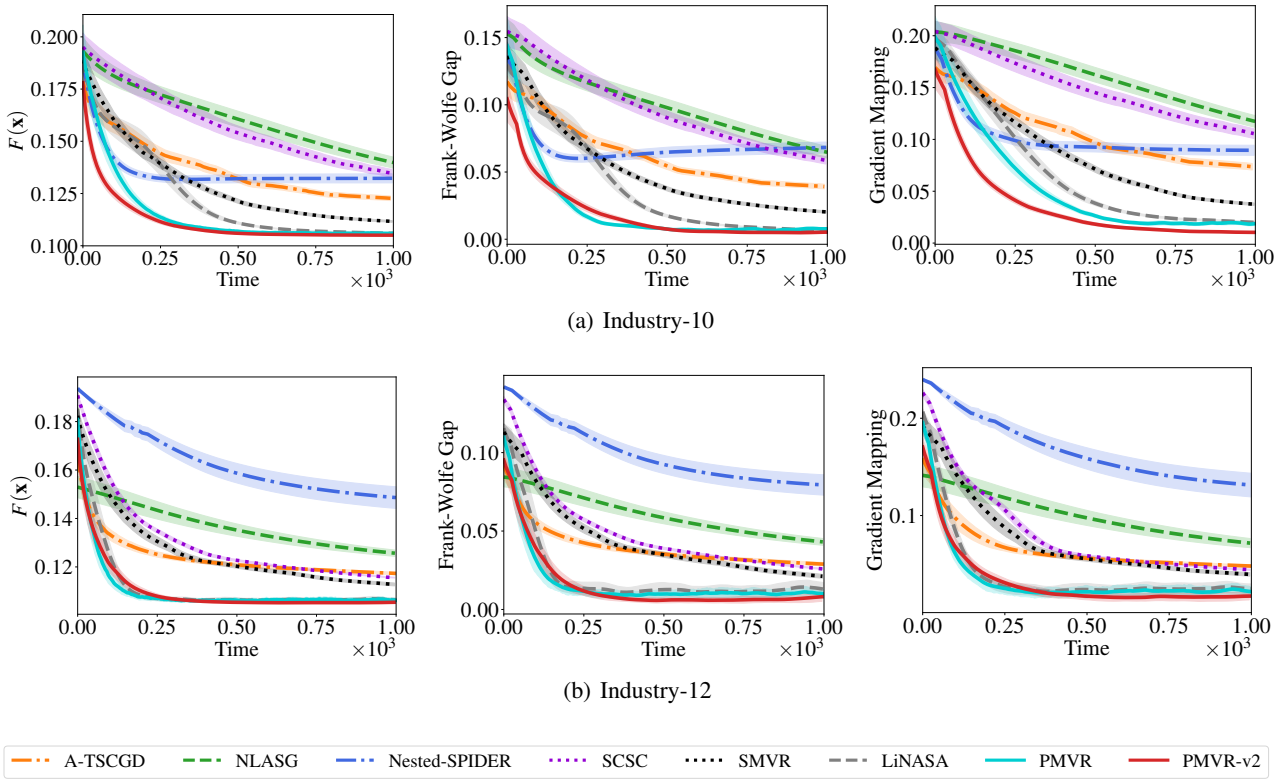


Figure 2. Results for risk-averse portfolio optimization.

4.2. Mean-variance Risk-averse Optimization

We first consider the problem of risk-averse portfolio optimization (Shapiro et al., 2021), a commonly used benchmark in comparing various multi-level algorithms (Yang et al., 2019; Balasubramanian et al., 2021; Zhang & Xiao, 2021; Chen et al., 2021). Suppose we have d assets to invest over time steps $\{1, \ldots, T\}$, and $\mathbf{r}_t \in \mathbb{R}^d$ represents the payoff of d assets at time step t. The goal is to maximize investment returns and minimize risk simultaneously. A suitable approach for this purpose is the mean-variance risk-averse optimization model, where risk is defined as the variance. This optimization problem is formulated as:

$$\min_{\mathbf{x}\in\mathcal{X}} F(\mathbf{x}) = -\frac{1}{T}\sum_{t=1}^{T}\langle \mathbf{r}_t, \mathbf{x}\rangle + \frac{\lambda}{T}\sum_{t=1}^{T}\left(\langle \mathbf{r}_t, \mathbf{x}\rangle - \langle \bar{\mathbf{r}}, \mathbf{x}\rangle\right)^2,$$

where $\bar{\mathbf{r}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{r}_t$, and the decision variable $\mathbf{x}$ denotes the investment quantities in d assets. The domain $\mathcal{X}$ is a simplex, ensuring that $\|\mathbf{x}\|_1 = 1$ for any $\mathbf{x} \in \mathcal{X}$. This problem can be modeled as a stochastic two-level constrained compositional optimization problem, with each layer expressed as follows:

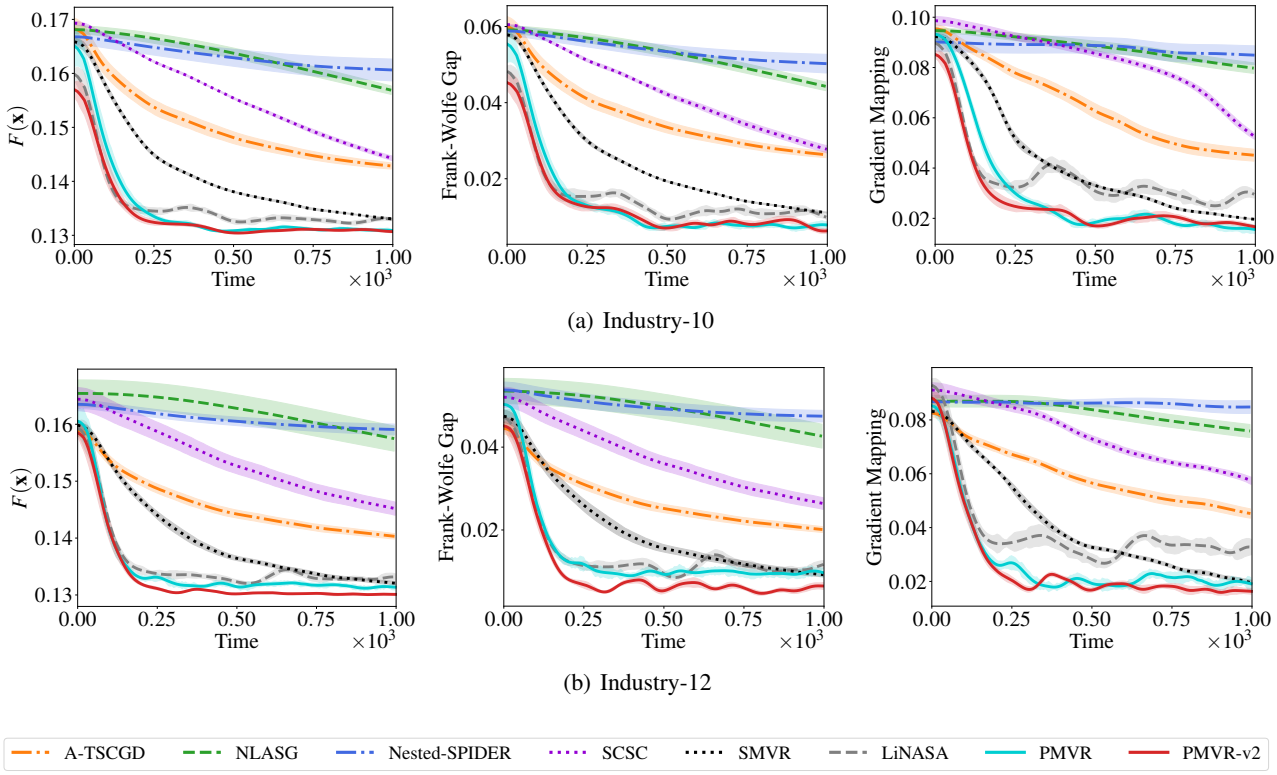
$$f_1(\mathbf{x}) = \left(-\frac{1}{T}\sum_{t=1}^{T}\langle \mathbf{r}_t, \mathbf{x}\rangle, \mathbf{x}\right),$$

$$f_2(\mathbf{y}_1, \mathbf{y}_2) = \mathbf{y}_1 + \frac{\lambda}{T}\sum_{t=1}^{T}\left(\langle \mathbf{r}_t, \mathbf{y}_2\rangle + \mathbf{y}_1\right)^2,$$

where $f_1(\cdot)$ is the inner function and $f_2(\cdot)$ is the outer function such that $F(\mathbf{x}) = f_2(f_1(\mathbf{x}))$.

For experimental validation, we utilize real-world datasets Industry-10 and Industry-12 from the Kenneth R. French Data Library[2]. These datasets include payoffs for 10 and 12 industrial assets over 25,105 consecutive periods. For non-projection-free methods, we implement the projection onto the simplex following a well-known efficient projection method (Duchi et al., 2008). We report the loss value $F(\mathbf{x})$, as well as the Frank-Wolfe gap and gradient mapping criteria in Figure 2, averaging all curves over 50 runs. We also include results concerning the number of iterations in Figure 4 in the Appendix A. It is observed that LiNASA+ICG, our PMVR and PMVR-v2, tend to converge more rapidly compared to other algorithms in all tasks. Specifically, PMVR

[2]https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/



(a) Industry-10

(b) Industry-12

Figure 3. Results for risk-averse portfolio optimization.

demonstrates similar performance to LiNASA+ICG in the Industry 12 dataset and performs better than LiNASA+ICG in Industry 10. The loss value, Frank-Wolfe gap, and gradient mapping of our PMVR-v2 decrease most quickly in both datasets, validating the effectiveness of the proposed method.

4.3. Mean-deviation Risk-averse Optimization

Finally, we further conduct the experiment on a three-level compositional optimization problem. Here, we still consider the problem of risk-averse portfolio optimization as in the previous subsection, and the risk is quantified as the standard deviation this time. The mathematical formulation of this problem is:

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

where $\bar{\mathbf{r}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{r}_t$, $\mathcal{X}$ is the probability simplex, and the decision variable $\mathbf{x}$ denotes the investment quantity vector in the d assets. This is a three-level compositional optimization problem according to the analysis by Jiang et al. (2022b).

In the experiments, we also evaluated different methods on the real-world datasets Industry-10 and Industry-12. The results, including the loss value $F(\mathbf{x})$, the Frank-Wolfe gap, and the gradient mapping criteria, are reported in Figure 3. These results are averaged over 10 runs. As can be seen, our methods (PMVR and PMVR-v2) demonstrated faster convergence compared to other algorithms across all tasks. The loss value, Frank-Wolfe gap, and gradient mapping of our PMVR-v2 decreased most rapidly in both datasets, indicating the effectiveness of our proposed approach.

5. Conclusion

In this paper, we investigate projection-free algorithms for stochastic constrained multi-level optimization. Our proposed methods not only yield better results than previous approaches under the gradient mapping criterion but also provide guarantees for the Frank-Wolfe gap, an aspect previously absent in projection-free multi-level research. Additionally, we provide theoretical guarantees for convex and strongly convex objective functions, and validate the effectiveness of our proposed methods through numerical experiments.

Acknowledgements

This work was partially supported by NSFC (62122037, 61921006), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Postgraduate Research & Practice Innovation Program of Jiangsu Province (No. KYCX24_0231).

Impact Statement

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

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. and Ghadimi, S. Zeroth-order nonconvex stochastic optimization: Handling constraints, high dimensionality, and saddle points. *Foundations of Computational Mathematics*, 22:35–76, 2018.

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.

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.

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 Ruszczynski, 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. C., Shalev-Shwartz, S., Singer, Y., and Chandra, T. Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, 2008.

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.

Frank, M. and Wolfe, P. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

Gao, H. Stochastic multi-level compositional optimization algorithms over networks with level-independent convergence rate. *ArXiv e-prints*, arXiv:2306.03322, 2023.

Garber, D. and Kretzu, B. Projection-free online exp-concave optimization. *ArXiv e-prints*, arXiv:2302.04859, 2023.

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.

Hazan, E. and Kale, S. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 521–528, 2012.

Hazan, E. and Luo, H. Variance-reduced and projection-free stochastic optimization. In *Proceedings of the 33rd International Conference on Machine Learning*, pp. 1263–1271, 2016.

Hazan, E. and Minasyan, E. Faster projection-free online learning. In *Proceedings of the 33rd Annual Conference on Learning Theory*, pp. 1877–1893, 2020.

Jaggi, M. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 427–435, 2013.

Ji, K., Yang, J., and Liang, Y. Multi-step model-agnostic meta-learning: Convergence and improved algorithms. *ArXiv e-prints*, arXiv:2002.07836, 2020.

Jiang, W., Li, G., Wang, Y., Zhang, L., and Yang, T. Multi-block-single-probe variance reduced estimator for coupled compositional optimization. In *Advances in Neural Information Processing Systems 35*, 2022a.

Jiang, W., Wang, B., Wang, Y., Zhang, L., and Yang, T. Optimal algorithms for stochastic multi-level compositional optimization. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 10195–10216, 2022b.

Jiang, W., Qin, J., Wu, L., Chen, C., Yang, T., and Zhang, L. Learning unnormalized statistical models via compositional optimization. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 15105–15124, 2023.

Lacoste-Julien, S. Convergence rate of frank-wolfe for non-convex objectives. *ArXiv e-prints*, arXiv:1607.00345, 2016.

Lan, G. and Zhou, Y. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

Li, T., Beirami, A., Sanjabi, M., and Smith, V. On tilted losses in machine learning: Theory and applications. *ArXiv e-prints*, arXiv:2109.06141, 2021.

Mhammedi, Z. Exploiting the curvature of feasible sets for faster projection-free online learning. *ArXiv e-prints*, arXiv:2205.11470, 2022.

Nesterov, Y. A method for solving the convex programming problem with convergence rate $\mathcal{O}(1/k^2)$. *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Qi, Q., Guo, Z., Xu, Y., Jin, R., and Yang, T. An online method for a class of distributionally robust optimization with non-convex objectives. *ArXiv e-prints*, arXiv:2006.10138, 2021.

Qu, C., Li, Y., and Xu, H. Non-convex conditional gradient sliding. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 4208–4217, 2018.

Reddi, S. J., Sra, S., Póczos, B., and Smola, A. Stochastic Frank-Wolfe methods for nonconvex optimization. *Annual Allerton Conference on Communication, Control, and Computing*, pp. 1244–1251, 2016.

Shapiro, A., Dentcheva, D., and Ruszczynski, A. *Lectures on Stochastic Programming: Modeling and Theory*. Society for Industrial and Applied Mathematics, 2021.

Wan, Y. and Zhang, L. Projection-free online learning over strongly convex sets. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 10076–10084, 2021.

Wan, Y., Tu, W.-W., and Zhang, L. Projection-free distributed online convex optimization with $\mathcal{O}(\sqrt{T})$ communication complexity. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9818–9828, 2020.

Wan, Y., Xue, B., and Zhang, L. Projection-free online learning in dynamic environments. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 10067–10075, 2021.

Wan, Y., Wang, G., Tu, W.-W., and Zhang, L. Projection-free distributed online learning with sublinear communication complexity. *Journal of Machine Learning Research*, 23(172):1–53, 2022.

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, Y., Yang, W., Jiang, W., Lu, S., Wang, B., Tang, H., Wan, Y., and Zhang, L. Non-stationary projection-free online learning with dynamic and adaptive regret guarantees. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pp. 15671–15679, 2024.

Xiao, T., Balasubramanian, K., and Ghadimi, S. A projection-free algorithm for constrained stochastic multi-level composition optimization. In *Advances in Neural Information Processing Systems 35*, pp. 19984–19996, 2022.

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.

Yang, Z., Balasubramanian, K., and Liu, H. High-dimensional non-Gaussian single index models via thresholded score function estimation. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 3851–3860, 2017.

Yu, D., Cai, Y., Jiang, W., and Zhang, L. Efficient algorithms for empirical group distributional robust optimization and beyond. *ArXiv e-prints*, arXiv:2403.03562, 2024.

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.

Yurtsever, A., Sra, S., and Cevher, V. Conditional gradient methods via stochastic path-integrated differential estimator. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 7282–7291, 2019.

Zhang, J. and Xiao, L. A stochastic composite gradient method with incremental variance reduction. In *Advances*

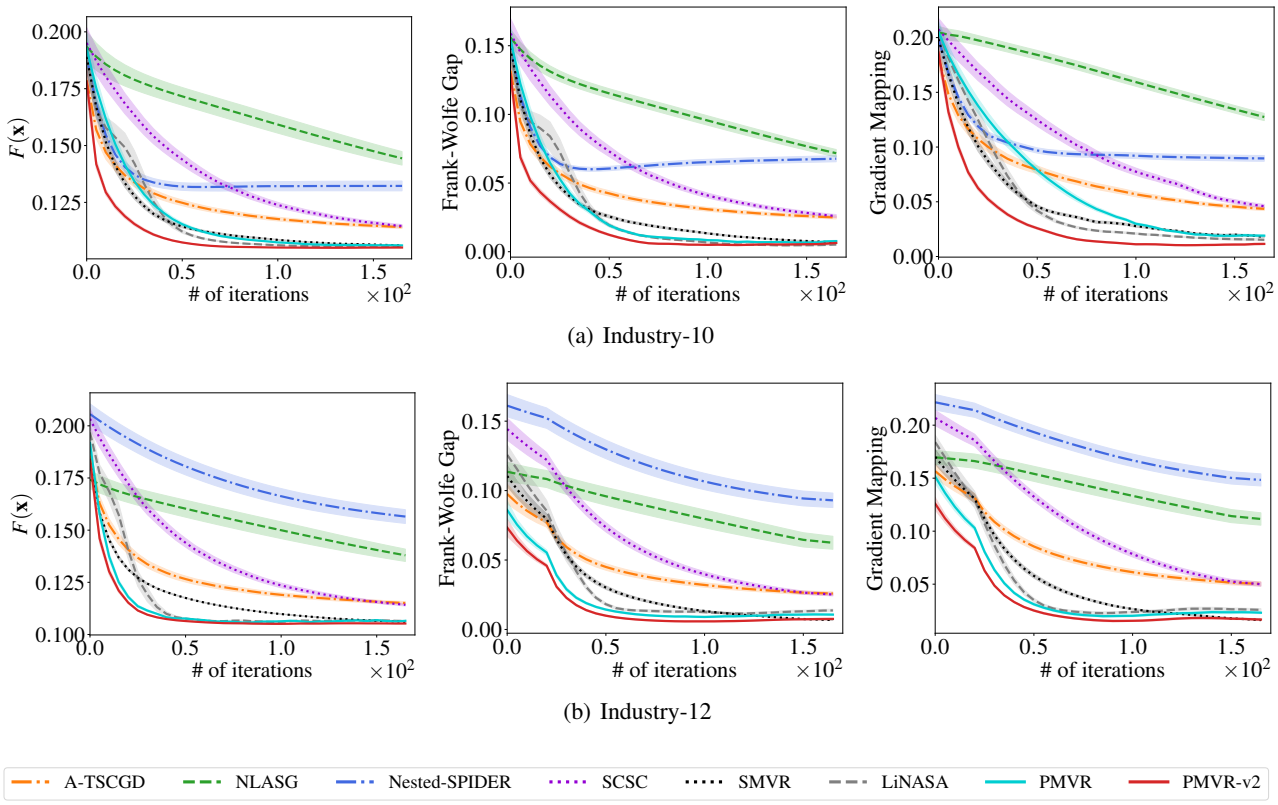
in Neural Information Processing Systems 33, pp. 9075–9085, 2019.

Zhang, J. and Xiao, L. Multilevel composite stochastic optimization via nested variance reduction. *SIAM Journal on Optimization*, 31(2):1131–1157, 2021.

Zhang, M., Shen, Z., Mokhtari, A., Hassani, H., and Karbasi, A. One sample stochastic frank-wolfe. In *International Conference on Artificial Intelligence and Statistics*, 2019.

A. More Experimental Results

We also report the loss value $F(\mathbf{x})$, as well as the Frank-Wolfe gap and gradient mapping criteria concerning the number of iterations for the mean-variance risk-averse optimization in Figure 4. When the cost of the projection operation is not considered (since we report the results based on iteration numbers rather than the time used), the SMVR, LiNASA+ICG, our PMVR and PMVR-v2, tend to converge more rapidly compared to other algorithms in all tasks. The loss value, Frank-Wolfe gap, and gradient mapping of our PMVR-v2 decrease most quickly in both datasets, validating the effectiveness of the proposed method.



(a) Industry-10

(b) Industry-12

Figure 4. Results for risk-averse portfolio optimization.

B. Proof of Frank-Wolfe Gap (Theorem 1 and Theorem 2)

In this section, we present the proofs for the Frank-Wolfe gap. First, to bound the estimation error of the gradient, we have the following lemma.

Lemma 1. *The gradient estimation error can be divided into the following two parts.*

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

Proof. First, it is easy to show that:

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

Then, we define that $\mathbf{y}_t^i = f_i \circ f_{i-1} \circ \ldots \circ f_1(\mathbf{w}_t)$ and $\nabla \widehat{F}_i(\mathbf{w}_t) = \nabla f_1(\mathbf{w}_t) \cdots \nabla f_i(\mathbf{u}_t^{i-1})$.

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

Besides, we also have:

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

To this end, we can conclude that:

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

where $C_i \coloneqq L_f^{K-1} L_J (1 + L_f + \ldots + L_f^{K-i-1})$. Since $C_i \leq L_F$, we know that:

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

□

Next, we bound the term $\mathbb{E}\left[\left\|\mathbf{v}_t - \prod_{i=1}^{K} \nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]$ and $\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right]$ separately.

Lemma 2. *The gradient estimator $\mathbf{v}_t$ enjoys the following guarantee:*

$$\begin{aligned}
\sum_{t=1}^{T} \mathbb{E}\left[\left\|\mathbf{v}_t - \prod_{i=1}^{K} \nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \leq{}& \frac{\mathbb{E}\left[\left\|\mathbf{v}_1 - \prod_{i=1}^{K} \nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right]}{\alpha} + \frac{2K^2 \alpha T \sigma_J^2}{B_1} L_f^{2K-2} \\
&+ \frac{2K}{\alpha B_1} \mathcal{L}_J^2 L_f^{2K-2} \sum_{t=1}^{T} \sum_{i=1}^{K} \mathbb{E}\left[\left\|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\right\|^2\right]
\end{aligned}$$

Proof. According to the definition of $\mathbf{v}_t$, we know that:

$$\mathbf{v}_t = \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j}) + (1-\alpha_t)\left(\mathbf{v}_{t-1} - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right).$$

Then we can obtain the following guarantee:

$$\begin{aligned}
&\left\|\mathbf{v}_t - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2 \\
=&\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j}) + (1-\alpha_t)\left(\mathbf{v}_{t-1} - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2 \\
=&\left\|(1-\alpha_t)\left(\mathbf{v}_{t-1} - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right) + \alpha_t\left(\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right)\right. \\
&\left. + \left(\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1}) - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) + \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\right\|^2
\end{aligned}$$

Since the expectation of last two terms equals zero, such that

$$\begin{aligned}
E\left[\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right] = 0,& \\
\mathbb{E}\left[\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1}) - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) + \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right] = 0,&
\end{aligned}$$

we would have that:

$$\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{v}_t - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \\
\leq&(1-\alpha)^2\mathbb{E}\left[\left\|\mathbf{v}_{t-1} - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right] + \mathbb{E}\left[\left\|\alpha\left(\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right)\right.\right. \\
&\left.\left. + \left(\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1}) + \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right)\right\|^2\right] \\
\leq&(1-\alpha_t)\mathbb{E}\left[\left\|\mathbf{v}_{t-1} - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right] + 2\alpha_t^2\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right] \\
&+ 2\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - \frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j}) - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1}) + \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]
\end{aligned}$$

Then, we would bound the last two terms, respectively. First, we have that:

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
=&\frac{1}{B_1^2}\sum_{j=1}^{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
=&\frac{1}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
=&\frac{1}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_{t-1}^{0})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right.\right.\\
&\qquad+\nabla f_1(\mathbf{u}_{t-1}^{0})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_{t-1}^{0})\nabla f_2(\mathbf{u}_{t-1}^{1})\prod_{i=3}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\\
&\qquad\cdots\\
&\qquad\left.\left.+\left(\prod_{i=1}^{K-1}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right)\nabla f_K(\mathbf{u}_{t-1}^{K-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
\le&\frac{K}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_{t-1}^{0})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right\|^2\right]\\
&+\frac{K}{B_1}\mathbb{E}\left[\left\|\nabla f_1(\mathbf{u}_{t-1}^{0})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_{t-1}^{0})\nabla f_2(\mathbf{u}_{t-1}^{1})\prod_{i=3}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right\|^2\right]\\
&+...\\
&+\frac{K}{B_1}\mathbb{E}\left[\left\|\left(\prod_{i=1}^{K-1}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right)\nabla f_K(\mathbf{u}_{t-1}^{K-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
\le&\frac{K}{B_1}\sigma_J^2(L_f^{2K-2}+L_f^{2K-2}+...+L_f^{2K-2})\\
=&\frac{K^2}{B_1}\sigma_J^2L_f^{2K-2}
\end{aligned}
$$

When dealing with the second term, note that

$$
\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\left(\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})+\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t}^{i-1})\right)\right\|\right]=0
$$

So, we can have that:

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\frac{1}{B_1}\sum_{j=1}^{B_1}\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})+\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\\
=&\frac{1}{B_1^2}\mathbb{E}\left[\left\|\sum_{j=1}^{B_1}\left(\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})+\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right)\right\|^2\right]\\
=&\frac{1}{B_1^2}\sum_{j=1}^{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})+\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\\
=&\frac{1}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})+\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\\
\leq&\frac{1}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right\|^2\right]\\
=&\frac{1}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right.\right.\\
&\quad+\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\nabla f_2(\mathbf{u}_t^1;\xi_t^{2,j})\prod_{i=3}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\\
&\quad\cdots\\
&\quad\left.\left.+\left(\prod_{i=1}^{K-1}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\nabla f_K(\mathbf{u}_{t-1}^{K-1};\xi_t^{K,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right\|^2\right]\\
\leq&\frac{K}{B_1}\mathbb{E}\left[\left\|\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right\|^2\right]\\
&+\frac{K}{B_1}\mathbb{E}\left[\left\|\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\prod_{i=2}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-\nabla f_1(\mathbf{u}_t^0;\xi_t^{1,j})\nabla f_2(\mathbf{u}_t^1;\xi_t^{2,j})\prod_{i=3}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})\right\|^2\right]\\
&\cdots\\
&+\frac{K}{B_1}\mathbb{E}\left[\left\|\left(\prod_{i=1}^{K-1}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\nabla f_K(\mathbf{u}_{t-1}^{K-1};\xi_t^{i,j})-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right\|^2\right]\\
\leq&\frac{K}{B_1}\mathcal{L}_J^2L_f^{2K-2}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1}-\mathbf{u}_{t-1}^{i-1}\right\|^2\right]
\end{aligned}
$$

To this end, we can conclude that:

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{v}_t-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\\
\leq&(1-\alpha_t)\mathbb{E}\left[\left\|\mathbf{v}_{t-1}-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]+\frac{2K^2\alpha_t^2\sigma_J^2}{B_1}L_f^{2K-2}+\frac{2K}{B_1}\mathcal{L}_J^2L_f^{2K-2}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1}-\mathbf{u}_{t-1}^{i-1}\right\|^2\right],
\end{aligned}
$$

which implies that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{v}_t-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\leq\sum_{t=2}^{T}\left(\frac{1}{\alpha_{t+1}}-\frac{1}{\alpha_t}\right)\mathbb{E}\left[\left\|\mathbf{v}_1-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right]$$
$$+\frac{1}{\alpha_2}\mathbb{E}\left[\left\|\mathbf{v}_1-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right]+\frac{2\sigma_J^2K^2L_f^{2K-2}}{B_1}\sum_{t=1}^{T}\alpha_{t+1}+\frac{2K}{B_1}\mathcal{L}_J^2L_f^{2K-2}\sum_{t=1}^{T}\frac{1}{\alpha_{t+1}}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1}-\mathbf{u}_{t-1}^{i-1}\right\|^2\right]$$

Note that $1/\alpha_{t+1}-1/\alpha_t=t^{1/2}-(t-1)^{1/2}\leq 1/2$ and for $t\geq 2$, $\sum_{t=1}^{T}t^{-2/3}\leqslant 3T^{1/3}+1$. As a result, we have that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{v}_t-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\leq$$
$$4\mathbb{E}\left[\left\|\mathbf{v}_1-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right]+\frac{16\sigma_J^2K^2L_f^{2K-2}}{B_1}T^{1/3}+\frac{4K}{B_1}\mathcal{L}_J^2L_f^{2K-2}\sum_{t=1}^{T}\frac{1}{\alpha_{t+1}}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1}-\mathbf{u}_{t-1}^{i-1}\right\|^2\right]$$

□

Lemma 3. *The inner function estimator $\mathbf{u}_t$ ensures that:*

$$\sum_{t=1}^{T}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^i-f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\leq\frac{\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_1^i-f_i(\mathbf{u}_1^{i-1})\right\|^2\right]}{\alpha}+\frac{2\alpha KT\sigma^2}{B_1}+\frac{2\mathcal{L}_f^2}{\alpha B_1}\sum_{t=1}^{T}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1}-\mathbf{u}_{t-1}^{i-1}\right\|^2\right]$$

Proof. Since $\mathbf{u}_t^i=\frac{1}{B_1}\sum_{j=1}^{B_1}f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})+(1-\alpha)(\mathbf{u}_{t-1}^i-\frac{1}{B_1}\sum_{j=1}^{B_1}f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}))$, we have:

$$\begin{aligned}
&\mathbb{E}\left[\left\|f_i(\mathbf{u}_t^{i-1})-\mathbf{u}_t^i\right\|^2\right]\\
=&\mathbb{E}\left[\left\|(1-\alpha)\left(\mathbf{u}_{t-1}^i-f_i(\mathbf{u}_{t-1}^{i-1})\right)+\frac{\alpha}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_{t-1}^{i-1})\right)\right.\right.\\
&\qquad\left.\left.+f_i(\mathbf{u}_{t-1}^{i-1})-f_i(\mathbf{u}_t^{i-1})-\frac{1}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\right\|^2\right]\\
\leq&(1-\alpha)^2\mathbb{E}\left\|\mathbf{u}_{t-1}^i-f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2+\frac{2\alpha^2}{B_1}\mathbb{E}\left[\left\|f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2\right]\\
&\quad+2\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_{t-1}^{i-1})+f_i(\mathbf{u}_t^{i-1})\right)\right\|^2\right]
\end{aligned}$$

where the last inequality is due to: $\mathbb{E}\left[f_i(\mathbf{u}_{t-1}^{i-1})-f_i(\mathbf{u}_t^{i-1})-\frac{1}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\right]=0$, as well as $\mathbb{E}\left[\frac{\alpha}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j})-f_i(\mathbf{u}_{t-1}^{i-1})\right)\right]=0$.

Also, since we know the fact that $\mathbb{E}\left[f_i(\mathbf{u}_{t-1}^{i-1}) - f_i(\mathbf{u}_t^{i-1}) - f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right] = 0$, we have:

$$\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{B_1}\sum_{j=1}^{B_1}\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_{t-1}^{i-1}) + f_i(\mathbf{u}_t^{i-1})\right)\right\|^2\right] \\
\leq&\frac{1}{B_1^2}\sum_{j=1}^{B_1}\mathbb{E}\left[\left\|\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_{t-1}^{i-1}) + f_i(\mathbf{u}_t^{i-1}) - f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\right\|^2\right] \\
\leq&\frac{1}{B_1^2}\sum_{j=1}^{B_1}\mathbb{E}\left[\left\|\left(f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right)\right\|^2\right] \\
\leq&\frac{1}{B_1}\mathbb{E}\left[\left\|f_i(\mathbf{u}_{t-1}^{i-1};\xi_t^{i,j}) - f_i(\mathbf{u}_t^{i-1};\xi_t^{i,j})\right\|^2\right] \\
\leq&\frac{1}{B_1}\mathcal{L}_f^2\left\|\mathbf{u}_{t-1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2.
\end{aligned}$$

As a result, we can conclude that:

$$\mathbb{E}\left[\left\|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\right\|^2\right] \leq (1-\alpha)\mathbb{E}\left\|\mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2 + \frac{2\alpha^2\sigma^2}{B_1} + \frac{2}{B_1}\mathcal{L}_f^2\left\|\mathbf{u}_{t-1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2$$

This leads to the fact that:

$$\sum_{i=1}^K\mathbb{E}\left[\left\|f_i(\mathbf{u}_t^{i-1}) - \mathbf{u}_t^i\right\|^2\right] \leq (1-\alpha)\sum_{i=1}^K\mathbb{E}\left\|\mathbf{u}_{t-1}^i - f_i(\mathbf{u}_{t-1}^{i-1})\right\|^2 + \frac{2\alpha^2\sigma^2 K}{B_1} + \frac{2}{B_1}\mathcal{L}_f^2\sum_{i=1}^K\left\|\mathbf{u}_{t-1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2$$

By summing up and rearranging, we can get:

$$\sum_{t=1}^T\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \leq \frac{\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_1^i - f_i(\mathbf{u}_1^{i-1})\right\|^2\right]}{\alpha} + \frac{2\alpha K T\sigma^2}{B_1} + \frac{2\mathcal{L}_f^2}{\alpha B_1}\sum_{t=1}^T\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\right\|^2\right],$$

which finishes the proof. □

Then, we try to bound the term $\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2\right]$.

Lemma 4.

$$\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2\right] \leq \left(\sum_{i=1}^K\left(2\mathcal{L}_f^2\right)^{i-1}\right)\left(\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2\right] + \frac{2\alpha^2\sigma^2 K}{B_1} + 2\alpha^2 K\sum_{i=1}^K\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\right)$$

Proof. We discuss the following two cases, separately.

1. For the first level, i.e., $i = 1$, we have:

$$\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1} - \mathbf{u}_t^{i-1}\right\|^2\right] = \mathbb{E}\left[\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2\right] = \mathbb{E}\left[\eta^2\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2\right].$$

2. For other levels, i.e., $2 \leq i \leq K$, we have:

$$\begin{aligned}
&\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1}-\mathbf{u}_{t}^{i-1}\right\|^{2}\right] \\
=&\mathbb{E}\left[\left\|\alpha\left(f_{i-1}(\mathbf{u}_{t}^{i-2})-\mathbf{u}_{t}^{i-1}\right)+\frac{1}{B_1}\sum_{j=1}^{B_1}(f_{i-1}(\mathbf{u}_{t+1}^{i-2};\xi_{t+1}^{i-1})-f_{i-1}(\mathbf{u}_{t}^{i-2};\xi_{t+1}^{i-1}))\right.\right. \\
&\left.\left.+\alpha\left(\frac{1}{B_1}\sum_{j=1}^{B_1}f_{i-1}(\mathbf{u}_{t}^{i-2};\xi_{t+1}^{i-1})-f_{i-1}(\mathbf{u}_{t}^{i-2})\right)\right\|^{2}\right] \\
\leq&2\mathbb{E}\left[\left\|\alpha\left(f_{i-1}(\mathbf{u}_{t}^{i-2})-\mathbf{u}_{t}^{i-1}\right)+\alpha\left(\frac{1}{B_1}\sum_{j=1}^{B_1}f_{i-1}(\mathbf{u}_{t}^{i-2};\xi_{t+1}^{i-1})-f_{i-1}(\mathbf{u}_{t}^{i-2})\right)\right\|^{2}\right] \\
&+2\mathcal{L}_f^2\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-2}-\mathbf{u}_{t}^{i-2}\right\|^{2}\right] \\
\leq&2\alpha^2\left\|f_{i-1}(\mathbf{u}_{t}^{i-2})-\mathbf{u}_{t}^{i-1}\right\|^{2}+\frac{2\alpha^2\sigma^2}{B_1}+2\mathcal{L}_f^2\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-2}-\mathbf{u}_{t}^{i-2}\right\|^{2}\right].
\end{aligned}$$

Denote $\Upsilon_t^i=\mathbb{E}\left[\left\|\mathbf{u}_t^i-f_i(\mathbf{u}_t^{i-1})\right\|^2\right]$ and $Q^i=\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1}-\mathbf{u}_t^{i-1}\right\|^2\right]$, we have $Q^i\leq 2\mathcal{L}_f^2Q^{i-1}+2\alpha^2\Upsilon_t^{i-1}+\frac{2\alpha^2\sigma^2}{B_1}$ for $i\geq 2$. Then we can get:

$$\begin{aligned}
Q^1&\leq\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right] \\
Q^2&\leq\left(2\mathcal{L}_f^2\right)\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2}{B_1}+2\alpha^2\Upsilon_t^1 \\
Q^3&\leq\left(2\mathcal{L}_f^2\right)^2\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2\left(1+2\mathcal{L}_f^2\right)}{B_1}+2\alpha^2\left(2\mathcal{L}_f^2\Upsilon_t^1+\Upsilon_t^2\right) \\
&\cdots \\
Q^i&\leq\left(2\mathcal{L}_f^2\right)^{i-1}\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2}{B_1}\sum_{j=1}^{i-1}\left(2\mathcal{L}_f^2\right)^{j-1}+2\alpha^2\sum_{j=1}^{i-1}\left(2\mathcal{L}_f^2\right)^{i-1-j}\Upsilon_t^j \\
&\leq\left(2\mathcal{L}_f^2\right)^{i-1}\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2}{B_1}\sum_{j=1}^{K}\left(2\mathcal{L}_f^2\right)^{j-1}+2\alpha^2\sum_{j=1}^{K}\sum_{l=1}^{K}\left(2L_f^2\right)^{K-l}\Upsilon_t^j.
\end{aligned}$$

When summing up, we have:

$$\begin{aligned}
\sum_{i=1}^{K}Q^i&\leq\sum_{i=1}^{K}\left(2\mathcal{L}_f^2\right)^{i-1}\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2K}{B_1}\sum_{i=1}^{K}\left(2\mathcal{L}_f^2\right)^{i-1}+2\alpha^2K\sum_{j=1}^{K}\sum_{l=1}^{K}\left(2L_f^2\right)^{K-l}\Upsilon_t^j \\
&\leq\left(\sum_{i=1}^{K}\left(2\mathcal{L}_f^2\right)^{i-1}\right)\left(\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2K}{B_1}+2\alpha^2K\sum_{i=1}^{K}\Upsilon_t^i\right)
\end{aligned}$$

So we have :

$$\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_{t+1}^{i-1}-\mathbf{u}_t^{i-1}\right\|^2\right]\leq\left(\sum_{i=1}^{K}\left(2\mathcal{L}_f^2\right)^{i-1}\right)\left(\mathbb{E}\left[\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2\right]+\frac{2\alpha^2\sigma^2K}{B_1}+2\alpha^2K\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^i-f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\right)$$

□

Lemma 5. *Denote that the constants as* $L_1 = \left(2K\mathcal{L}_J^2 L_f^{2K-2} + 4\mathcal{L}_f^2\right)\left(\sum_{i=1}^{K}\left(2\mathcal{L}_f^2\right)^{i-1}\right)$, $L_2 = \max\{2, 2KL_F^2\}$, $L_3 = 2L_2(2K\sigma_J^2 L_f^{2K-2} + 4K\sigma^2 + 2L_1\sigma^2 K + L_1 D^2 + L_1)$*, we can ensure that:*

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2\right] \leq \frac{L_3}{\alpha B_0} + \frac{\alpha L_3 T}{B_1} + \frac{L_3\eta^2 T}{\alpha B_1}.$$

Proof. Based on Lemma 1, we have:

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

Noting that $\mathbb{E}\left[\left\|\mathbf{v}_1 - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right] \leq \frac{K^2\sigma_J^2 L_f^{2K-2}}{B_0}$, $\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \leq \frac{K\sigma^2}{B_0}$ and setting $2\alpha L_1 K \leq B_1$, we can deduce that:

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{v}_t - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right] + 2\sum_{t=1}^{T}\sum_{i=1}^{K-1}\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \\
\leq&\frac{K^2\sigma_J^2 L_f^{2K-2} + 2K\sigma^2}{\alpha B_0} + \frac{2\alpha K^2 T\sigma_J^2 L_f^{2K-2} + 4\alpha KT\sigma^2}{B_1} + \frac{2K\mathcal{L}_J^2 L_f^{2K-2} + 4\mathcal{L}_f^2}{\alpha B_1}\sum_{t=1}^{T}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^{i-1} - \mathbf{u}_{t-1}^{i-1}\right\|^2\right] \\
\leq&\frac{K^2\sigma_J^2 L_f^{2K-2} + 2K\sigma^2}{\alpha B_0} + \frac{2\alpha K^2 T\sigma_J^2 L_f^{2K-2} + 4\alpha KT\sigma^2}{B_1} \\
&+ \frac{L_1}{\alpha B_1}\left(T\eta^2 D^2 + \frac{2T\alpha^2\sigma^2 K}{B_1} + 2\alpha^2 K\sum_{t=1}^{T}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right]\right) \\
\leq&\frac{K^2\sigma_J^2 L_f^{2K-2} + 2K\sigma^2}{\alpha B_0} + \frac{2\alpha K^2 T\sigma_J^2 L_f^{2K-2} + 4\alpha KT\sigma^2}{B_1} + \frac{L_1\eta^2 D^2 T}{\alpha B_1} + \frac{2\alpha L_1\sigma^2 KT}{B_1^2} \\
&+ \sum_{t=1}^{T}\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right]
\end{aligned}$$

So, we have that:

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2\right] \\
\leq&L_2\sum_{t=1}^{T}\mathbb{E}\left[\left\|\mathbf{v}_t - \prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2\right] + L_2\sum_{t=1}^{T}\sum_{i=1}^{K-1}\mathbb{E}\left[\left\|\mathbf{u}_t^i - f_i(\mathbf{u}_t^{i-1})\right\|^2\right] \\
\leq&L_2\left(\frac{K^2\sigma_J^2 L_f^{2K-2} + 2K\sigma^2}{\alpha B_0} + \frac{2\alpha K^2 T\sigma_J^2 L_f^{2K-2} + 4\alpha KT\sigma^2}{B_1} + \frac{L_1\eta^2 D^2 T}{\alpha B_1} + \frac{2\alpha L_1\sigma^2 KT}{B_1^2}\right) \\
\leq&\frac{L_3}{\alpha B_0} + \frac{\alpha L_3 T}{B_1} + \frac{L_3\eta^2 T}{\alpha B_1}
\end{aligned}$$

□

Now we can finish the proof as follows. Denote the Frank-Wolfe Gap as $\mathcal{F}(\mathbf{x}) := \max_{\hat{\mathbf{x}}\in\mathcal{X}}\langle\hat{\mathbf{x}} - \mathbf{x}, -\nabla F(\mathbf{x})\rangle$ and $\mathbf{z}_t^\star = \arg\max_{\hat{\mathbf{x}}\in\mathcal{X}}\langle\hat{\mathbf{x}} - \mathbf{x}, -\nabla F(\mathbf{x})\rangle$.

$$
\begin{aligned}
F\left(\mathbf{x}_{t+1}\right) &\leq F\left(\mathbf{x}_t\right)+\left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{x}_{t+1}-\mathbf{x}_t\right\rangle+\frac{L_F}{2}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 \\
&= F\left(\mathbf{x}_t\right)+\eta\left\langle \mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta\left\langle \nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta^2 \frac{L_F}{2} D^2 \\
&\leq F\left(\mathbf{x}_t\right)+\eta\left\langle \mathbf{v}_t, \mathbf{z}_t^{\star}-\mathbf{x}_t\right\rangle+\eta\left\langle \nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta^2 \frac{L_F}{2} D^2 \\
&= F\left(\mathbf{x}_t\right)+\eta\left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{z}_t^{\star}-\mathbf{x}_t\right\rangle+\eta\left\langle \nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t, \mathbf{z}_t-\mathbf{z}_t^{\star}\right\rangle+\eta^2 \frac{L_F}{2} D^2 \\
&\leq F\left(\mathbf{x}_t\right)-\eta \mathcal{F}\left(\mathbf{x}_t\right)+\eta D\left\|\nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t\right\|+\eta^2 \frac{L_F}{2} D^2
\end{aligned}
$$

That is to say:

$$
\frac{1}{T} \sum_{t=1}^T \mathcal{F}\left(\mathbf{x}_t\right) \leq \frac{F\left(\mathbf{x}_1\right)-F\left(\mathbf{x}_{T+1}\right)}{\eta T}+\frac{D}{T} \sum_{t=1}^T\left\|\nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t\right\|+\eta \frac{L_F}{2} D^2
$$

By setting $T=\mathcal{O}\left(\epsilon^{-2}\right), \eta=\mathcal{O}(\epsilon), \alpha=\mathcal{O}(\epsilon), B_0=B_1=\mathcal{O}\left(\epsilon^{-1}\right)$, we can ensure that $\mathcal{F}(\mathbf{x}) \leq \epsilon$. Moreover, we can obtain the same guarantee by setting that $\alpha=\mathcal{O}\left(\epsilon^2\right), \eta=\mathcal{O}\left(\epsilon^2\right), B_1=\mathcal{O}(1), B_0=\mathcal{O}\left(\epsilon^{-1}\right), T=\mathcal{O}\left(\epsilon^{-3}\right)$.

C. Proof of Gradient Mapping (Theorem 3 and Theorem 4)

In the previous analysis of Lemma 5, we simply reduce $\eta^2\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2 \leq \eta^2 D^2$. To obtain the optimal rate, we have to keep this term. That is to say, we rewrite the Lemma 5 as follows

$$
\frac{1}{T} \sum_{t=1}^T \mathbb{E}\left[\left\|\nabla F(\mathbf{x}_t)-\mathbf{v}_t\right\|^2\right] \leq \frac{L_3}{\alpha B_0 T}+\frac{\alpha L_3}{B_1}+\frac{\eta^2 L_3}{\alpha B_1 T} \sum_{t=1}^T\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2 .
$$

According to Proposition 2 of Xiao et al. (2022), we know that the gradient mapping

$$
\|\mathcal{G}(\mathbf{x}_t, \beta)\|^2 \leq-4 \beta g(\mathbf{x}_t, \mathbf{v}_t)+2 \mathbb{E}\left[\left\|\nabla F(\mathbf{x}_t)-\mathbf{v}_t\right\|^2\right],
$$

where $g(\mathbf{x}_t, \mathbf{v}_t)=\min_{\mathbf{y} \in \mathcal{X}}\left\{\langle \mathbf{v}_t, \mathbf{y}-\mathbf{x}\rangle+\frac{\beta}{2}\|\mathbf{y}-\mathbf{x}_t\|^2\right\}$.

Due to the convergence of Frank-Wolfe algorithm (Jaggi, 2013), we know that

$$
\left\langle \mathbf{v}_t, \mathbf{z}_t-\mathbf{x}\right\rangle+\frac{\beta}{2}\|\mathbf{z}_t-\mathbf{x}_t\|^2 \leq g(\mathbf{x}_t, \mathbf{v}_t)+\frac{2 \beta D^2}{N+2},
$$

which is widely used in the analysis of Frank-Wolfe algorithm (Xiao et al., 2022; Zhang et al., 2019; Wan et al., 2021; Wan & Zhang, 2021). Now, we begin our proof

$$
\begin{aligned}
& F\left(\mathbf{x}_{t+1}\right) \\
& \leq F\left(\mathbf{x}_t\right)+\left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{x}_{t+1}-\mathbf{x}_t\right\rangle+\frac{L_F}{2}\left\|\mathbf{x}_{t+1}-\mathbf{x}_t\right\|^2 \\
& \leq F\left(\mathbf{x}_t\right)+\eta\left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta^2 \frac{L_F}{2}\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2 \\
& = F\left(\mathbf{x}_t\right)+\eta\left\langle \mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta\left\langle \nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\eta^2 \frac{L_F}{2}\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2 \\
& = F\left(\mathbf{x}_t\right)+\eta\left\langle \mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle+\frac{\eta \beta}{2}\|\mathbf{z}_t-\mathbf{x}_t\|^2+\eta\left\langle \nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t, \mathbf{z}_t-\mathbf{x}_t\right\rangle-\frac{\eta \beta}{2}\|\mathbf{z}_t-\mathbf{x}_t\|^2+\eta^2 \frac{L_F}{2}\left\|\mathbf{z}_t-\mathbf{x}_t\right\|^2
\end{aligned}
$$

Denote $\mathbf{y}^\star = \min_{\mathbf{y}\in\mathcal{X}} \left\{ \langle \mathbf{v}_t, \mathbf{y}-\mathbf{x}\rangle + \frac{\beta}{2}\|\mathbf{y}-\mathbf{x}_t\|^2 \right\}$. Set $\eta \leq \frac{\beta}{2L_F}$ (note that β is a positive constant), and then we have,

$$\begin{aligned}
F(\mathbf{x}_{t+1}) &\leq F(\mathbf{x}_t) + \eta\langle \mathbf{v}_t, \mathbf{y}^\star - \mathbf{x}_t\rangle + \frac{\eta\beta}{2}\|\mathbf{y}^\star - \mathbf{x}_t\|^2 + \frac{2\eta\beta D^2}{N+2} \\
&\quad + \eta\langle \nabla F(\mathbf{x}_t) - \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t\rangle - \frac{\eta\beta}{2}\|\mathbf{z}_t - \mathbf{x}_t\|^2 + \eta^2\frac{L_F}{2}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\leq F(\mathbf{x}_t) + \eta g(\mathbf{x}_t, \mathbf{v}_t) + \frac{2\beta D^2\eta}{N+2} + \eta\langle \nabla F(\mathbf{x}_t) - \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t\rangle - \frac{\eta\beta}{2}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\quad + \eta^2\frac{L_F}{2}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\leq F(\mathbf{x}_t) + \eta g(\mathbf{x}_t, \mathbf{v}_t) + \frac{2\beta D^2\eta}{N+2} + \eta\langle \nabla F(\mathbf{x}_t) - \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t\rangle - \frac{\eta\beta}{4}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\leq F(\mathbf{x}_t) + \eta g(\mathbf{x}_t, \mathbf{v}_t) + \frac{2\beta D^2\eta}{N+2} + \frac{2\eta}{\beta}\mathbb{E}\left[\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\|^2\right] - \frac{\eta\beta}{8}\|\mathbf{z}_t - \mathbf{x}_t\|^2
\end{aligned}$$

As a result,

$$-g(\mathbf{x}_t, \mathbf{v}_t) \leq \frac{F(\mathbf{x}_t) - F(\mathbf{x}_{t+1})}{\eta} + \frac{2\beta D^2}{N+2} + \frac{2}{\beta}\mathbb{E}\left[\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\|^2\right] - \frac{\beta}{8}\|\mathbf{z}_t - \mathbf{x}_t\|^2$$

So, we have:

$$\begin{aligned}
&\|\mathcal{G}(\mathbf{x}_t, \beta)\|^2 \\
\leq& -4\beta g(\mathbf{x}_t, \mathbf{v}_t) + 2\mathbb{E}\left[\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\|^2\right] \\
\leq& \frac{4\beta(F(\mathbf{x}_t) - F(\mathbf{x}_{t+1}))}{\eta} + \frac{8\beta^2 D^2}{N+2} + 10\mathbb{E}\left[\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\|^2\right] - \frac{\beta^2}{2}\|\mathbf{z}_t - \mathbf{x}_t\|^2
\end{aligned}$$

Finally, we have

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\|\mathcal{G}(\mathbf{x}_t, \beta)\|^2 \\
&\leq \frac{4\beta\Delta_F}{\eta T} + \frac{8\beta^2 D^2}{N} + \frac{10}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\|^2\right] - \frac{\beta^2}{2}\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\leq \frac{4\beta\Delta_F}{\eta T} + \frac{8\beta^2 D^2}{N} + \frac{10L_3}{\alpha B_0 T} + \frac{10L_3\alpha}{B_1} + \left(10L_3\frac{\eta^2}{\alpha B_1} - \frac{\beta^2}{2}\right)\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{z}_t - \mathbf{x}_t\|^2 \\
&\leq \frac{4\beta\Delta_F}{\eta T} + \frac{8\beta^2 D^2}{N} + \frac{10L_3}{\alpha B_0 T} + \frac{10L_3\alpha}{B_1}
\end{aligned}$$

The last inequality holds with $\eta \leq \sqrt{\frac{\beta^2\alpha B_1}{20L_3}}$. By setting $\alpha = \mathcal{O}(\sqrt{\epsilon})$, $\eta = \mathcal{O}(1)$, $T = \mathcal{O}(\epsilon^{-1})$, $B_0 = \mathcal{O}(\epsilon^{-0.5})$, $B_1 = \mathcal{O}(\epsilon^{-0.5})$, and $N = \mathcal{O}(\epsilon^{-1})$, We can ensure that $\mathbb{E}\left[\|\mathcal{G}(\mathbf{x}_t, \beta)\|^2\right] \leq \epsilon$. This guarantee can also be satisfied by setting $\alpha = \mathcal{O}(\epsilon), \eta = \mathcal{O}(\sqrt{\epsilon}), T = \mathcal{O}\left(\epsilon^{-1.5}\right), B_0 = \mathcal{O}(\epsilon^{-0.5}), B_1 = \mathcal{O}(1), N = \mathcal{O}(\epsilon^{-1})$.

D. Proof of Optimal Gap (Theorem 5, 6, 7 and 8)

In this section, we investigate the optimal gap for convex and strongly convex objective functions.

D.1. Convex

According to the equation (C.21) of Yurtsever et al. (2019), for algorithm 1 with convex objectives, we have that

$$\mathbb{E}\left[F(\mathbf{x}_{t+1})\right] - F_\star \leq (1-\eta)\left(\mathbb{E}\left[F(\mathbf{x}_t)\right] - F_\star\right) + \eta D\mathbb{E}\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\| + \eta^2\frac{L_F D^2}{2}$$

Then we have:

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[F\left(\mathbf{x}_{t+1}\right)\right]-F_{\star}\\
\leq&\frac{\left(\mathbb{E}\left[F\left(\mathbf{x}_1\right)\right]-F_{\star}\right)}{\eta T}+\frac{D}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t\right\|+\eta\frac{L_F D^2}{2}\\
\leq&\frac{\left(\mathbb{E}\left[F\left(\mathbf{x}_1\right)\right]-F_{\star}\right)}{\eta T}+D\sqrt{\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left\|\nabla F\left(\mathbf{x}_t\right)-\mathbf{v}_t\right\|^2}+\eta\frac{L_F D^2}{2}\\
\leq&\frac{\left(\mathbb{E}\left[F\left(\mathbf{x}_1\right)\right]-F_{\star}\right)}{\eta T}+D\sqrt{\frac{\mathbb{E}\left[L_2\left\|\mathbf{v}_1-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_1^{i-1})\right\|^2\right]+2L_2\sum_{i=1}^{K}\mathbb{E}\left[\left\|\mathbf{u}_1^i-f_i(\mathbf{u}_1^{i-1})\right\|^2\right]}{\alpha T}}\\
&+D\sqrt{\frac{\alpha L_3}{B_1}+\frac{\eta^2 L_3}{\alpha B_1}}+\eta\frac{L_F D^2}{2}
\end{aligned}$$

Next, we denote that $\Gamma_s=\frac{1}{T}\sum_{t=1}^{T}\left\|\mathbf{v}_t-\prod_{i=1}^{K}\nabla f_i(\mathbf{u}_t^{i-1})\right\|^2+\frac{2}{T}\sum_{t=1}^{T}\sum_{i=1}^{K}\left\|\mathbf{u}_t^i-f_i(\mathbf{u}_t^{i-1})\right\|^2$ for stage s and we denote $\mathbf{x}^s$ as the output of Algorithm 3 for the stage s. Then, we have:

$$\mathbb{E}\left[F\left(\mathbf{x}^s\right)-F_{\star}\right]\leq\frac{\mathbb{E}\left[F\left(\mathbf{x}^{s-1}\right)-F_{\star}\right]}{\eta_s T_s}+D\sqrt{\frac{L_2\Gamma_{s-1}}{\alpha_s T_s}}+D\sqrt{\frac{\alpha_s L_3}{B_1^s}+\frac{\eta_s^2 L_3}{\alpha_s B_1^s}}+\eta_s\frac{L_F D^2}{2}$$

Also, we have that:

$$\Gamma_s\leq\frac{2\Gamma_{s-1}}{\alpha_s T_s}+\frac{\alpha_s L_3}{B_1^s}+\frac{L_3\eta_s^2}{\alpha_s B_1^s}$$

Set $\epsilon_s=\left(\frac{1}{2}\right)^{s-1}$, $\eta_s=\alpha_s\leq\frac{\epsilon_s}{4L_F D^2}, T_s\geq\max\left\{\frac{(4\Delta_F+24+64D^2L_2)}{\eta_s},\frac{\Gamma_0(16D^2L_2+6)}{\eta_s}\right\}, B_1^s\geq\max\left\{\frac{12\eta_s L_3}{\epsilon_s^2},\frac{128L_3\eta_s D^2}{\epsilon_s^2}\right\}$. We can guarantee that $\mathbb{E}\left[F\left(\mathbf{x}^s\right)-F_{\star}\right]\leq\epsilon_s$ and $\mathbb{E}\left[\Gamma_s\right]\leq\epsilon_s^2$. We will use the induction to give the proof:

Proof. When $s=1$, we have:

$$\begin{aligned}
\mathbb{E}\left[F\left(\mathbf{x}^1\right)-F_{\star}\right]&\leq\frac{\Delta_F}{\eta_1 T_1}+D\sqrt{\frac{L_2\Gamma_0}{\alpha_1 T_1}}+D\sqrt{\frac{\alpha_1 L_3}{B_1^1}+\frac{\eta_1^2 L_3}{\alpha_1 B_1^1}}+\eta_1\frac{L_F D^2}{2}\\
&\leq\epsilon_1=1
\end{aligned}$$

where $\Gamma_0=K^2\sigma_J^2 L_f^{2K-2}+2K\sigma^2$. Also, we have that:

$$\Gamma_1\leq\frac{2\Gamma_0}{\alpha_1 T_1}+\frac{\alpha_1 L_3}{B_1^1}+\frac{L_3\eta_1^2}{\alpha_1 B_1^1}\leq\epsilon_1^2=1$$

Then, assume $\mathbb{E}\left[F\left(\mathbf{x}^s\right)-F_{\star}\right]\leq\epsilon_s$ and $\mathbb{E}\left[\Gamma_s\right]\leq\epsilon_s^2$, we would prove that it holds for stage $s+1$.

$$\begin{aligned}
&\mathbb{E}\left[F\left(\mathbf{x}^{s+1}\right)-F_{\star}\right]\\
&\leq\frac{\epsilon_s}{\eta_{s+1}T_{s+1}}+D\sqrt{\frac{L_2\epsilon_s^2}{\alpha_{s+1}T_{s+1}}}+D\sqrt{\frac{\alpha_{s+1}L_3}{B_1^{s+1}}+\frac{\eta_{s+1}^2L_3}{\alpha_{s+1}B_1^{s+1}}}+\eta_{s+1}\frac{L_F D^2}{2}\\
&\leq\frac{\epsilon_s}{8}+\frac{\epsilon_s}{8}+\frac{\epsilon_s}{8}+\frac{\epsilon_s}{8}=\epsilon_{s+1}
\end{aligned}$$

We also know that

$$\begin{aligned}\Gamma_{s+1} \leq & \frac{2\epsilon_s^2}{\alpha_{s+1}T_{s+1}} + \frac{\alpha_{s+1}L_3}{B_1^{s+1}} + \frac{L_3\eta_{s+1}^2}{\alpha_{s+1}B_1^{s+1}} \\ \leq & \frac{\epsilon_s^2}{12} + \frac{\epsilon_s^2}{12} + \frac{\epsilon_s^2}{12} = \epsilon_{s+1}^2\end{aligned}$$

So we prove $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \left(\frac{1}{2}\right)^{s-1}$ with $\eta_s = \alpha_s \leq \frac{\epsilon_s}{4L_F D^2}, B_1^s \geq \max\left\{\frac{12\eta_s L_3}{\epsilon_s^2}, \frac{128L_3\eta_s D^2}{\epsilon_s^2}\right\}$, $T_s \geq \max\left\{\frac{(4\Delta_F+24+64D^2L_2)}{\eta_s}, \frac{\Gamma_0(16D^2L_2+6)}{\eta_s}\right\}$. This condition can be satisfied by setting that $\eta_s = \mathcal{O}(\epsilon_s)$, $T_s = \mathcal{O}(\epsilon_s^{-1})$, $\alpha = \mathcal{O}(\epsilon_s)$ and $B_1^s = \mathcal{O}(\epsilon_s^{-1})$ [Large batch version]. This can also be achieved by setting that $\eta_s = \mathcal{O}(\epsilon_s^2)$, $T_s = \mathcal{O}(\epsilon_s^{-2})$, $\alpha = \mathcal{O}(\epsilon_s^2)$ and $B_1^s = \mathcal{O}(1)$ [Constant Batch].

To ensure $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \epsilon$, set $S = \mathcal{O}(\log_2(\frac{1}{\epsilon}))$, and the SFO rate is $\sum_{s=1}^S T^s B_1^s = \mathcal{O}\left(\sum_{s=1}^S 2^{(2s)}\right) = \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$. □

D.2. Strongly Convex

In this section, we assume $F(\mathbf{x})$ is λ-strongly convex function and we set $\eta \leq \frac{\lambda}{4L_F}$:

$$\begin{aligned}& F\left(\mathbf{x}_{t+1}\right) \\ \leq & F\left(\mathbf{x}_t\right) + \left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{x}_{t+1} - \mathbf{x}_t \right\rangle + \frac{L_F}{2}\left\|\mathbf{x}_{t+1} - \mathbf{x}_t\right\|^2 \\ \leq & F\left(\mathbf{x}_t\right) + \eta\left\langle \nabla F\left(\mathbf{x}_t\right), \mathbf{z}_t - \mathbf{x}_t \right\rangle + \eta^2\frac{L_F}{2}\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2 \\ = & F\left(\mathbf{x}_t\right) + \eta\left\langle \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t \right\rangle + \eta\left\langle \nabla F\left(\mathbf{x}_t\right) - \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t \right\rangle + \frac{\eta\lambda}{4}\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2 + (\eta^2\frac{L_F}{2} - \frac{\eta\lambda}{4})\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2 \\ \leq & F(\mathbf{x}_t) + \eta\left\langle \nabla F(\mathbf{x}_t) - \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}^\star \right\rangle + \eta\left\langle \nabla F(\mathbf{x}_t), \mathbf{x}^\star - \mathbf{x}_t \right\rangle + \frac{\eta\lambda}{4}\left\|\mathbf{x}^\star - \mathbf{x}_t\right\|^2 + \frac{\eta\lambda D^2}{N} - \frac{\eta\lambda}{8}\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2,\end{aligned}$$

where the last inequality is due to the fact that

$$\eta\left\langle \mathbf{v}_t, \mathbf{z}_t - \mathbf{x}_t \right\rangle + \frac{\eta\lambda}{4}\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2 \leq \eta\left\langle \mathbf{v}_t, \mathbf{x}^\star - \mathbf{x}_t \right\rangle + \frac{\eta\lambda}{4}\left\|\mathbf{x}^\star - \mathbf{x}_t\right\|^2 + \frac{\eta\lambda D^2}{N}$$

For λ-strongly convex function, we have $\langle \nabla F(\mathbf{x}_t), \mathbf{x}^\star - \mathbf{x}_t\rangle \leq F_\star - F(\mathbf{x}_t) - \frac{\lambda}{2}\left\|\mathbf{x}_t - \mathbf{x}^\star\right\|^2$. As a result:

$$\begin{aligned}F\left(\mathbf{x}_{t+1}\right) \leq & F(\mathbf{x}_t) + \eta(F_\star - F(\mathbf{x}_t)) - \frac{\eta\lambda}{4}\left\|\mathbf{x}^\star - \mathbf{x}_t\right\|^2 + \frac{\eta\lambda}{16}\left\|\mathbf{z}_t - \mathbf{x}^\star\right\|^2 + \frac{4\eta}{\lambda}\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2 \\ & + \frac{\eta\lambda D^2}{N} - \frac{\eta\lambda}{8}\left\|\mathbf{z}_t - \mathbf{x}_t\right\|^2\end{aligned}$$

So we have:

$$F\left(\mathbf{x}_{t+1}\right) - F_\star \leq (1-\eta)(F\left(\mathbf{x}_t\right) - F_\star) + \frac{4\eta}{\lambda}\left\|\nabla F(\mathbf{x}_t) - \mathbf{v}_t\right\|^2 + \frac{\eta\lambda D^2}{N}$$

Finally, we have:

$$\frac{1}{T}\sum_{i=1}^T (F\left(\mathbf{x}_t\right) - F_\star) \leq \frac{F\left(\mathbf{x}_1\right) - F_\star}{\eta T} + \frac{4}{\lambda T}\sum_{t=1}^T\left\|\nabla F\left(\mathbf{x}_t\right) - \mathbf{v}_t\right\|^2 + \frac{\lambda D^2}{N}$$

Next, we denote $\mathbf{x}^s$ as the output for stage s. Then, we have:

$$\mathbb{E}\left[F\left(\mathbf{x}^s\right)\right] - F_\star \leq \frac{F\left(\mathbf{x}_{s-1}\right) - F_\star}{\eta_s T_s} + \frac{4\Gamma_{s-1}}{\lambda\alpha_s T_s} + \frac{4\alpha_s L_3}{\lambda B_1^s} + \frac{4\eta_s^2 L_3}{\lambda\alpha_s B_1^s} + \frac{\lambda D^2}{N}$$

Also, we have that:

$$\Gamma_s \leq \frac{2\Gamma_{s-1}}{\alpha_s T_s} + \frac{\alpha_s L_3}{B_1^s} + \frac{L_3 \eta_s^2}{\alpha_s B_1^s}$$

Set that $\epsilon_s = (\frac{1}{2})^{s-1}, B_1^s \geq \frac{72\alpha_s L_3}{\lambda \epsilon_s}, N \geq \frac{6\lambda D^2}{\epsilon_s}, T_s \geq \max\{\frac{72(\Delta_F+1)}{\eta_s}, \frac{72(D^2+1)\Gamma_0}{\eta_s}\}$, $B_0 = \max\{\lambda^{-1}, 1\}$. We can guarantee that $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \epsilon_s$ and $\mathbb{E}\left[\Gamma_s\right] \leq \lambda \epsilon_s$. We will use the induction to give the proof:

Proof. When $s = 1$, we have:

$$\Gamma_1 \leq \frac{2\Gamma_0}{\alpha_1 T_1 B_0} + \frac{\alpha_1 L_3}{B_1^1} + \frac{L_3 \eta_1^2}{\alpha_1 B_1^1} \leq \lambda \epsilon_1 = \lambda,$$

where B_0 denotes the batch size used only in the first iteration of the first stage. Also, we have that:

$$\begin{aligned}\mathbb{E}\left[F\left(\mathbf{x}^1\right) - F_\star\right] &\leq \frac{\Delta_F}{\eta_1 T_1} + \frac{4\Gamma_0}{\lambda \alpha_1 T_1 B_0} + \frac{4\alpha_1 L_3}{\lambda B_1^1} + \frac{4\eta_1^2 L_3}{\lambda \alpha_1 B_1^1} + \frac{\lambda D^2}{N} \\ &\leq \epsilon_1 = 1\end{aligned}$$

Then assume $\mathbb{E}\left[\Gamma_s\right] \leq \lambda \epsilon_s$ and $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \epsilon_s$, we would prove that it holds for stage $s+1$.

$$\begin{aligned}\Gamma_{s+1} \leq& \frac{2\Gamma_s}{\alpha_{s+1} T_{s+1}} + \frac{\alpha_{s+1} L_3}{B_1^{s+1}} + \frac{L_3 \eta_{s+1}^2}{\alpha_{s+1} B_1^{s+1}} \\ \leq& \frac{2\lambda \epsilon_s}{\alpha_{s+1} T_{s+1}} + \frac{\alpha_{s+1} L_3}{B_1^{s+1}} + \frac{L_3 \eta_{s+1}^2}{\alpha_{s+1} B_1^{s+1}} \\ \leq& \frac{\lambda \epsilon_s}{2} = \lambda \epsilon_{s+1}\end{aligned}$$

Besides, we know that

$$\begin{aligned}\mathbb{E}\left[F\left(\mathbf{x}^{s+1}\right) - F_\star\right] &\leq \frac{\epsilon_s}{\eta_{s+1} T_s} + \frac{4\Gamma_s}{\lambda \alpha_{s+1} T_{s+1}} + \frac{4\alpha_{s+1} L_3}{\lambda B_1^{s+1}} + \frac{4\eta_{s+1}^2 L_3}{\lambda \alpha_{s+1} B_1^{s+1}} + \frac{\lambda D^2}{N} \\ &\leq \frac{\epsilon_s}{2} = \epsilon_{s+1}\end{aligned}$$

□

So we prove $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \left(\frac{1}{2}\right)^{s-1}$ with $\epsilon_s = (\frac{1}{2})^{s-1}, B_1^s \geq \frac{72\alpha_s L_3}{\lambda \epsilon_s}, N \geq \frac{6\lambda D^2}{\epsilon_s}, T_s \geq \frac{72(\Delta_F+1+\Gamma_0)}{\eta_s}$, $B_0 = \max\{\lambda^{-1}, 1\}$.

This condition can be satisfied by setting that $\eta_s = \alpha_s = \mathcal{O}(\lambda)$, $T_s = \mathcal{O}(\lambda^{-1})$, $B_1 = \mathcal{O}(\epsilon_s^{-1}), N = \mathcal{O}(\frac{\lambda}{\epsilon_s})$ (Large Batch) or by setting that $\eta_s = \alpha_s = \mathcal{O}(\lambda \epsilon_s)$, $T_s = \mathcal{O}(\lambda^{-1} \epsilon_s^{-1})$, $B_1 = \mathcal{O}(1), N = \mathcal{O}(\frac{\lambda}{\epsilon_s})$ (Constant Batch). To ensure $\mathbb{E}\left[F\left(\mathbf{x}^s\right) - F_\star\right] \leq \epsilon$, set $S = \mathcal{O}(\log_2(\frac{1}{\epsilon}))$, and the SFO rate is $\sum_{s=1}^S T_s B_1^s = \sum_{s=1}^S \frac{\mathcal{O}(1)}{\lambda} 2^{s-1} = \mathcal{O}(\frac{1}{\lambda \epsilon})$.