Provable Stochastic Optimization for Global Contrastive Learning: Small Batch Does Not Harm Performance

Zhuoning Yuan [1] **Yuexin Wu** [2] **Zi-Hao Qiu** [3] **Xianzhi Du** [2] **Lijun Zhang** [3] **Denny Zhou** [2] **Tianbao Yang** [1]

Abstract

In this paper, we study contrastive learning from an optimization perspective, aiming to analyze and address a fundamental issue of existing contrastive learning methods that either rely on a large batch size or a large dictionary of feature vectors. We consider a global objective for contrastive learning, which contrasts each positive pair with all negative pairs for an anchor point. From the optimization perspective, we explain why existing methods such as SimCLR require a large batch size in order to achieve a satisfactory result. In order to remove such requirement, we propose a memory-efficient **S**tochastic **O**ptimization algorithm for solving the **G**lobal objective of **C**ontrastive **L**earning of **R**epresentations, named SogCLR. We show that its optimization error is negligible under a reasonable condition after a sufficient number of iterations or is diminishing for a slightly different global contrastive objective. Empirically, we demonstrate that SogCLR with small batch size (e.g., 256) can achieve similar performance as SimCLR with large batch size (e.g., 8192) on self-supervised learning task on ImageNet-1K. We also attempt to show that the proposed optimization technique is generic and can be applied to solving other contrastive losses, e.g., two-way contrastive losses for bimodal contrastive learning. The proposed method is implemented in our open-sourced library LibAUC (`www.libauc.org`).

1. Introduction

Recently, self-supervised learning (SSL) for pre-training deep neural networks, which springs from natural language processing (Mikolov et al., 2013; Devlin et al., 2018; Lan

et al., 2019), has emerged to be a popular paradigm in computer vision for learning visual representations (Dosovitskiy et al., 2020; Zhu et al., 2020c; Liu et al., 2021). A simple yet effective framework of SSL for learning visual representations is contrastive learning (Chopra et al., 2005; Chen et al., 2020a), which uses the gradient of a contrastive loss to update model, aiming to push the similarity scores between positive pairs (augmented data from the same image) to be higher than that between negative pairs (augmented data from different images).

While the great performance of contrastive learning methods and their alternatives have been demonstrated on popular benchmarks (e.g., ImageNet), some fundamental problems of contrastive learning remain unresolved. One such problem is the requirement for large batch size. Unlike supervised learning methods, the performance of SimCLR (Chen et al., 2020a) decreases as the batch size decreases, and a satisfactory performance can be only achieved with a large batch size on natural image datasets (e.g., 8192 for ImageNet). However, in practice, training models with such a large batch size can be memory-intensive and requires more computational resources, especially when adopting large-scale backbones (e.g., Vision Transformers (Dosovitskiy et al., 2021; Zhai et al., 2021)) or taking video sequences as input (Qian et al., 2021).

To address this issue, some ad-hoc approaches have been investigated. For example, the MoCo method (He et al., 2020) uses a large dictionary to maintain a set of feature vectors for constructing negative pairs with data in the mini-batch. Other approaches choose to get around such issue by optimizing pairwise loss (Grill et al., 2020; Zbontar et al., 2021; Chen & He, 2021) or other losses (Caron et al., 2020). Nevertheless, the fundamental issue of optimizing a contrastive loss with a large batch size requirement still exists. This also occurs in other tasks with a similar contrastive loss, e.g., bimodal SSL tasks by optimizing a two-way contrastive loss (e.g., CLIP (Radford et al., 2021)).

In this paper, we aim to address this fundamental problem from the optimization perspective by considering a global objective for contrastive learning, providing a rigorous analysis to explain why SimCLR requires a large mini-batch size, and designing a memory-efficient stochastic algorithm

---

for optimizing the global contrastive objective with provable convergence guarantee under a reasonable condition. Our major contributions are summarized below:

- We propose a global objective for contrastive learning, in which the similarity score between a random positive pair of an anchor point is contrasted with that between the anchor point and all other images and their augmented data. We cast the problem as a special case of coupled compositional stochastic optimization by highlighting the challenges in designing stochastic algorithms.

- We analyze SimCLR from the perspective of optimizing the global contrastive objective, and show that it suffers from an optimization error of SimCLR in the order of $O(1/\sqrt{B})$ for the objective's gradient norm even with the number of iterations approaching infinity, which explains the phenomena that SimCLR's performance degrades as the mini-batch size decreases.

- We propose a memory-efficient stochastic algorithm named SogCLR without relying on a large batch size. We establish the convergence of the proposed algorithm SogCLR and show that its optimization error for the aforementioned global contrastive objective is negligible under a mild condition. Moreover, we show that SogCLR converges to a stationary solution to a slightly different global contrastive objective with a diminishing optimization error as the number of iterations increases.

- We demonstrate the empirical success of SogCLR on ImageNet-1K. With a standard mini-batch size 256 and the same other settings as SimCLR, by running 800 epochs, SogCLR achieves a performance of 69.4% for top 1 linear evaluation accuracy, which is better than 69.3% of SimCLR using a large batch size 8,192. By incorporating other useful techniques into SogCLR, e.g., multi-crop data augmentation and multi-layer projection heads, we are able to achieve 72.5% top-1 linear evaluation accuracy on ImageNet-1K, which is competitive with existing listwise contrastive loss based SSL methods using a large dictionary (e.g., MoCo-v2). We further demonstrate the usefulness of the proposed technique for bimodal contrastive learning, e.g., CLIP.

Finally, we would like to emphasize that to the best of our knowledge, this is the first work that analyzes SimCLR and a stochastic algorithm for contrastive learning from an optimization perspective. We expect that this paper would inspire new studies by proposing better algorithms for optimizing the global contrastive objective.

2. Related Work

We would like to point out that SSL is an emerging field and there are tremendous studies proposing different method-ologies. Nevertheless, we focus our attention on different methodologies for contrastive SSL.

Contrastive Losses. There are multiple definitions of contrastive loss, including pairwise losses, and listwise losses. The notation of contrastive loss dates back to 15 years ago for dimensionality reduction (Hadsell et al., 2006), which uses pairwise contrastive losses that simply push the similarity scores between positive pairs to be high and that between negative pairs to be low. Listwise contrastive losses have been proposed in the context of distance metric learning (Sohn, 2016), which contrasts a similarity score between an anchor point and a positive sample with a number of similarity scores between the anchor point and multiple negative samples. (Oord et al., 2018) is a pioneering work that uses a contrastive loss for unsupervised representation learning. They propose a contrastive loss based on noise contrastive estimation (NCE), which is called InfoNCE. It was used to learn representations by predicting the future in the latent space by using autoregressive model. However, a fundamental issue regarding how to select the negative samples and how it affects the learning performance was not studied in (Oord et al., 2018).

Contrastive SSL. The InfoNCE loss was later adopted in the momentum contrast (MoCo) method (He et al., 2020) for SSL of visual representations. MoCo tackles the question of how to construct negative samples in the latent space. It introduces two techniques (i) a momentum encoder network, which is used to generate representations of images for contrast with that generated by the target network on the anchor points, and is updated by a momentum step; (ii) a large dictionary that stores a number of feature representations for constructing negative pairs that are generated by the momentum encoder network, and is updated by a queue structure in a FIFO fashion. Later, the large dictionary and momentum contrast was abandoned in SimCLR (Chen et al., 2020a), which uses a large batch to sample data for constructing positive and negative pairs within the batch. SimCLR makes several contributions for improving the performance, in particular using strong data augmentations and MLP projection layers. SimCLR conducted extensive experiments by studying how the batch size and other factors (e.g. number of epochs) affect the performance and a key observation is that the performance degrades as the mini-batch size decreases. Although a large-batch size is preferred or not an issue in industrial setting, the fundamental issue of requiring large batch size is still not well addressed.

Improvements on top of MoCo or SimCLR. Recently, there have been some efforts made for improving MoCo and SimCLR. MoCo-v2 (Chen et al., 2020c) is an improved version of MoCo by adopting strong data augmentations and multiple MLP projection heads as in SimCLR. MoCo-v2 (with ResNet-50 as backbone) achieves 71.1% top-1 linear

Table 1. Comparison of different InfoNCE-loss based contrastive learning methods and their top-1 linear evaluation accuracy by using 800 epochs, a batch size of 256, and ResNet-50 on ImageNet-1K. Momentum encoder is introduced by MoCo (He et al., 2020). We expect the performance of SogCLR can be further improved by incorporating other techniques, e.g., InfoMin augmentation.

Method	Batch Size	Memory Bank	Momentum Encoder	Other Tricks	Convergence	Top1 Acc.
SimCLR (Chen et al., 2020a)	Large-batch	No	No	Strong Aug.	No	66.5
NNCLR (Dwibedi et al., 2021)	Large-batch	No	No	Nearest Neighbors	No	68.7
SiMo (Zhu et al., 2020a)	Small-batch	No	Yes	Margin Trick	No	72.1
MoCov2 (Chen et al., 2020c)	Small-batch	Yes	Yes	Strong Aug.	No	71.1
InfoMin (Tian et al., 2020)	Small-batch	Yes	Yes	InfoMin Aug.	No	73.0
SogCLR (Ours)	Small-batch	No	No	GC Optimization	Yes	72.5

evaluation accuracy on ImageNet-1K with small batch size of 256, which suppresses the SimCLR's 69.3% with larger batch size of 8192.

Several works have tried to approach SSL from the Info-Max principle, i.e., maximizing mutual information between different views and the shifted negative InfoNCE loss is a lower bound of mutual information (Tian et al., 2020; Chen et al., 2021a; Zhu et al., 2020b). (Tian et al., 2020) studies the effects of data augmentations from the perspective of mutual information between different views, and uses more and stronger data augmentations on the top of MoCo-v2 framework, which achieves 73% top-1 linear evaluation accuracy on Imagenet-1K with ResNet-50. (Chen et al., 2021a) clarifies why InfoNCE loss fails under small batch-size settings by showing that the negative InfoNEC loss approaches saturation (e.g., $\log(B)$ where B is batch size) after a few epochs. To address this issue, they propose a self-normalized version of InfoNCE loss named FlatNCE, in which larger weights will prioritize harder negative samplers in a mini-batch to facilitate the models learning better representations. Zhu et al. (2020b) proposes a similar technique from the viewpoint of mutual information lower bound to address the small batch collapse issue. In detail, they add a constant margin to offset the similarity score between positive pairs, which could help increase the weight of learning from hard negative samples.

(Li et al., 2020; Dwibedi et al., 2021; Chuang et al., 2020) improve InfoNCE in a different way. For example, (Li et al., 2020) proposes ProtoNCE loss that uses prototypical representations in place of the second view in InfoNCE loss, which are learned by clustering the data into a large number of clusters. (Dwibedi et al., 2021) uses similar samples for a given image extracted from a support set maintained by a queue to improve the data diversity for training. (Chuang et al., 2020) proposes a debiased contrastive loss to tackle the sampling bias (false negative samples) and they observe sampling negative samples from the truly different labels improves the performance. We compare our method with several InfoNCE-loss based SSL methods in Table 1 with top-1 linear evaluation accuracy reported on ImageNet-1K.

Global Contrastive Loss. The global contrastive loss that contrasts a positive pair with all possible negative pairs has been explored in the literature. For example, Wang & Isola

(2020) formulates the InfoNCE loss with the number of negative samples approaching infinity into two parts and explain them from the perspectives of alignment and uniformity, where alignment aims to keep similar positive pairs closer and uniformity aims to perverse the maximal information among all pairs by pushing them evenly distributed on the hypersphere. However, The challenge of handling a large number of components in the normalization term in the InfoNCE loss is not well addressed. One of the most well-known techniques to address this challenge is to use NCE (Gutmann & Hyvärinen, 2010), which reduces the problem to how to sample negative data and how many negative data are sufficient for obtaining satisfactory performance. Although the original paper of NCE shows that the approximation error of NCE decreases as the number of samples increases, it is unclear how it affects the performance of SSL for visual representations. In contrast, this paper provides an arguable better approach for tackling the global contrastive loss in the sense that (i) the performance does not hinge on how to sample negative samples and how many to sample; (ii) there is a stronger convergence guarantee for optimizing the global contrastive loss.

Evolution of State-of-the-art. Caron et al. (2020) proposes SwAV that solves a swapped prediction problem wherein the prototypical codes obtained from one data augmented view are predicted using the similarity scores between the other view's representations and the prototypical codes. They also propose a multi-crop augmentation strategy, which significantly boosts the performance under small batch setting. In particular, SwAV achieves a top-1 linear evaluation accuracy of 75.3% using a batch size of 256 on ImageNet-1K. Grill et al. (2020) proposes a method named BYOL which minimizes a pairwise loss based on feature representations of two data augmented views from two neural networks, referred to as online and target networks, respectively. From an augmented view of an image, an online network is learned to predict the target network representation of the same image under a different augmented view. The target network is updated by a momentum step as in MoCo. BYOL achieves 74.3% top-1 linear evaluation accuracy on ImageNet-1K with a ResNet-50. Recently, a concurrent work (Tomasev et al., 2022) attains state-of-the-art performance of top-1 linear evaluation accuracy (77.1%) by using a combination of techniques, e.g., the InfoNCE loss as in SimCLR/MoCo, the

online-target setup as in MoCo/BYOL, the multi-crop augmentations as in SwAV, and the invariance loss as in (Mitrovic et al., 2020). We also notice that larger backbones lead to better performance, e.g, ResNet-200 (Grill et al., 2020; Tomasev et al., 2022) and Vision Transformer (Chen et al., 2021c; Li et al., 2021). We would like to point out that it is not our focus to leverage all of these different techniques to achieve state-of-the-art performance. But instead we focus on understanding the fundamental limits of optimizing the InfoNCE loss and providing an alternative yet effective strategy to make contrastive learning possible without using a large batch size, which is potentially useful for different methods, e.g., supervised contrastive learning (Khosla et al., 2020), and bimodal contrastive learning mentioned below (Radford et al., 2021).

Recently, a two-way contrastive loss has been used in bimodal contrastive learning (Zhang et al., 2021; Radford et al., 2021), which takes paired image and text as input and aims to map them into a Euclidean space that are closer than non-observed image/text pairs. The CLIP model (Radford et al., 2021) uses this idea to train a model on a large image-text dataset and observe promising zero-shot prediction performance for downstream tasks. However, their approach also uses a very large batch size equal to 32,768.

3. Optimizing Global Contrastive Objective

Notations. Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ denote the set of training images, let $\mathcal{P}$ denote a set of data augmentation operators that can be applied to each image to generate a copy. Let $\mathcal{A}(\cdot) \in \mathcal{P}$ denote a random data augmentation operator, and let $\mathbf{x} \in \mathcal{D}$ denote a random example from $\mathcal{D}$. Let $\mathcal{S}_i = \{\mathcal{A}(\mathbf{x}) : \forall \mathcal{A} \in \mathcal{P}, \forall \mathbf{x} \in \mathcal{D} \setminus \{\mathbf{x}_i\}\}$ denote all training images including their augmented versions but excluding that of $\mathbf{x}_i$. Let $E(\cdot)$ denote the encoder network parameterized by $\mathbf{w} \in \mathbb{R}^d$ that outputs a normalized feature representation of an input image. Below, $\mathcal{A}(\mathbf{x}_i)$ and $\mathcal{A}(\mathbf{x}_j)$ denote two independent random data augmentations applied to $\mathbf{x}_i$ and $\mathbf{x}_j$ independently.

The SimCLR method is to update the model according to the gradient of the local contrastive loss that is defined over sampled mini-batch data. To this end, a random mini-batch of B images $\mathcal{B} = \{\mathbf{x}_1, \cdots, \mathbf{x}_B\}$ are first sampled. Then for each image $\mathbf{x}_i \in \mathcal{B}$, two random augmented data $\mathcal{A}(\mathbf{x}_i), \mathcal{A}'(\mathbf{x}_i)$ are generated by two randomly sampled data augmentations $\mathcal{A}, \mathcal{A}' \in \mathcal{P}$. Then the gradient is computed based on the following local contrastive loss for each data $\mathbf{x}_i$ and its symmetric one by switching $\mathcal{A}$ and $\mathcal{A}'$:

$$L_{\mathcal{B}}(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}') = -\ln \frac{\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))/\tau)}{g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B})}, \quad (1)$$

where τ is known as the temperature parameter, and

$$g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) = \sum_{\mathbf{z}_j \in \mathcal{B}_i} (\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}_j)/\tau) \quad (2)$$

and $\mathcal{B}_i = \{\mathcal{A}(\mathbf{x}_j), \mathcal{A}'(\mathbf{x}_j) : \mathbf{x}_j \in \mathcal{B} \setminus \{\mathbf{x}_i\}\}$ denote the set of images that are generated by applying independent two random data augmentations to each image in $\mathcal{B}$ independently excluding $\mathbf{x}_i$.

3.1. A Global Contrastive Objective: V1

The local contrastive loss defined over the mini-batch samples hides the complexity for contrastive learning, which renders the SimCLR method sensitive to the mini-batch size. To address this issue, we propose a global contrastive objective. To this end, we define the following global contrastive loss for each augmented data pair $(\mathcal{A}(\mathbf{x}_i), \mathcal{A}'(\mathbf{x}_i))$:

$$L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}') = -\ln \frac{\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))/\tau)}{\varepsilon' + g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)}, \quad (3)$$

where $\varepsilon' > 0$ is a small constant, which is introduced simply for the purpose of analysis to ensure the denominator that involves $g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ is lower bounded [1], and

$$g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) = \sum_{\mathbf{z} \in \mathcal{S}_i} (\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z})/\tau), \quad (4)$$

which contrasts the similarity score between each positive pair $E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))$ with the similarity scores of negative pairs $E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z})$ for all $\mathbf{z} \in \mathcal{S}_i$. Based on the individual contrastive loss, we define the following **global contrastive objective (GCO)** for minimization:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \mathbb{E}_{\mathbf{x}_i \sim \mathcal{D}, \mathcal{A}, \mathcal{A}' \sim \mathcal{P}}[\tau L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}')] \quad (5)$$

where $\sim$ denotes a random sample, L is multiplied by τ to ensure the gradient is not illy scaled. In contrast to another variant proposed in section 3.4, we refer to the above objective as the V1 GCO.

To highlight the challenge for optimizing the global contrastive objective, we consider the calculation of the gradient of $\tau L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}')$ in terms of the parameters $\mathbf{w}$ of the encoder network E.

$$\tau \nabla L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}') = -\nabla(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i)))$$
$$+ \frac{\tau}{\varepsilon' + g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i).$$

It is notable that the first term can be easily computed by back-propogation. The challenge lies at computing the second term, where $g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ involves a large number of examples in $\mathcal{S}_i$ that includes all images and their augmented data excluding that of $\mathbf{x}_i$. Due to the finite-sum structure of g in (4), we can compute an unbiased estimator by sampling data from $\mathcal{S}_i$. Indeed, we can show that $\frac{1}{|\mathcal{B}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)$ is an unbiased estimator of $\frac{1}{|\mathcal{S}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$. SimCLR directly uses this mini-batch

---

[1]We can also modify the definition of $\mathcal{S}_i$ to include $\mathcal{A}(\mathbf{x}_i)$ for ensuring $g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ is lower bounded without adding ε'.

estimator to estimate $g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, S_i)$ and $\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ in the above equation, yielding the following approximated gradient of $\tau L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}')$:

$$\tau \widehat{\nabla} L(\mathbf{x}_i, \mathcal{A}, \mathcal{A}') = -\nabla(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))) \quad (6)$$
$$+ \frac{\tau}{\varepsilon + g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i),$$

where $\varepsilon = \frac{|\mathcal{B}_i|\varepsilon'}{|\mathcal{S}_i|}$. However, this quantity is a biased estimator of $\tau \nabla L(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{A}')$ due to the non-linear function $\frac{1}{\varepsilon + g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)}$.

3.2. SimCLR and its Convergence for V1 GCO

The SimCLR method can be viewed as a mini-batch based stochastic method, which uses a gradient estimator that is the average of the estimator in (6) for $\mathbf{x}_i$ in the sampled mini-batch. To analyze the optimization error of SimCLR, we first consider the following simplest update [2]:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \widehat{\nabla} L(\mathbf{x}_i, \mathcal{A}, \mathcal{A}'). \quad (7)$$

We establish the optimization error of the above update for T iterations for optimizing the V1 GCO.

Theorem 1. *Assume F is smooth, g is smooth and Lipchitiz continuous, SimCLR with the update (7) ensures that $\mathbb{E}[\|\nabla F(\mathbf{w}_{t'})\|^2] \leq O(\frac{1}{\eta T} + \eta + \frac{1}{B})$ for a random $t' \in \{1, \ldots, T\}$.*

Remark: The above theorem implies that SimCLR suffers an optimization error at least in the order of $O(1/\sqrt{B})$ for the objective's gradient norm. Even with $T \to \infty$, its optimization error is always dominated by $O(1/\sqrt{B})$. This explains the phenomenon that the performance of SimLCR degrades as the mini-batch size decreases. The above theorem also implies in order to find an ϵ-level stationary solution, i.e., $\mathbb{E}[\|\nabla F(\mathbf{w}_t')\| \leq \epsilon]$, we can set $\eta = O(\epsilon^2)$ and $T = O(1/\epsilon^4)$ and $B = O(1/\epsilon^2)$. All missing proofs can be found in the supplement.

3.3. SogCLR and its Convergence for V1 GCO

To address the issue of SimCLR, in this section we propose a memory-efficient stochastic algorithm for solving (5) without suffering from a large optimization error depending on the batch size. To this end, we decompose the objective function into three terms:

$$F(\mathbf{w}) = \mathbb{E}_{\mathbf{x}_i \sim \mathcal{D}, \mathcal{A}, \mathcal{A}' \sim \mathcal{P}}(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))) \quad (8)$$
$$+ \frac{\tau}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_{\mathcal{A}} \ln \left(\frac{\varepsilon'}{|\mathcal{S}_i|} + \frac{1}{|\mathcal{S}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \right) + \text{Const},$$

where Const is a constant that is independent of the model parameters. Below, we let $f(\cdot) = \tau \ln(\varepsilon'/|\mathcal{S}_i| + \cdot)$.

---

[2]For simplicity, we do not include another similar term in the gradient estimator by switching $\mathcal{A}$ and $\mathcal{A}'$, which will not affect the analysis.

Algorithm 1 SogCLR

1: **Input:** $\mathbf{w}_0 \in \mathbb{R}^d$, Initialize $\mathbf{u}_0 \in \mathbb{R}^n$
2: Draw a batch of B samples denoted by $\mathcal{B} = \{\mathbf{x}_i\}_{i=1}^B$.
3: **for** $t = 1, \ldots, T$ **do**
4:      **for** $\mathbf{x}_i \in \mathcal{B}$ **do**
5:          Compute $g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)$ and $g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}', \mathcal{B}_i)$ according to (2)
6:          Update $\mathbf{u}_{i,t}$ according to (9)
7:      **end for**
8:      Compute the gradient estimator $\mathbf{m}_t$ by (10)
9:      $\mathbf{v}_t = (1 - \beta)\mathbf{v}_{t-1} + \beta\mathbf{m}_t$
10:      $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta\mathbf{v}_t$ (or use Adam-style update)
11: **end for**

Our algorithm is motivated by the coupled compositional stochastic optimization studied in (Qi et al., 2021) for maximizing Average Precision, whose objective has a form of $\frac{1}{n} \sum_i f(g_i(\mathbf{w}))$ that is similar to the second component in our objective $F(\mathbf{w})$. The key idea of the proposed algorithm is to keep track of $\frac{1}{|\mathcal{S}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ by a scalar, whose averaged error in the long run is diminishing. However, different from the problem studied in (Qi et al., 2021), there could be many data augmentations in $\mathcal{P}$. As a result, by maintaining a scalar for each $\mathbf{x}_i \in \mathcal{D}, \mathcal{A} \in \mathcal{P}$, the memory cost is $O(n|\mathcal{P}|)$ which increases as we increase the number of data augmentations and could be very large if $|\mathcal{P}|$ is large. By noting that $\mathcal{A}(\mathbf{x}_i), \forall \mathcal{A} \in \mathcal{P}$ is an augmented data from the same image for different $\mathcal{A}$, we expect that their embedded feature vectors are close in the sense that $\mathbb{E}_{\mathcal{A}, \mathcal{A}', \mathbf{z}} |E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}) - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})|^2 \leq \epsilon^2$ for any $\mathcal{A}, \mathcal{A}', \mathbf{x}_i$ and a small value ϵ. By leveraging this property, we maintain and update a scalar $\mathbf{u}_i$ for each image to track $\frac{1}{|\mathcal{S}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$. At the t-th iteration, we update $\mathbf{u}_i$ for $\mathbf{x}_i \in \mathcal{B}$ by moving average

$$\mathbf{u}_{i,t} = (1 - \gamma)\mathbf{u}_{i,t-1}$$
$$+ \gamma \frac{1}{2|\mathcal{B}_i|}(g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) + g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}', \mathcal{B}_i)), \quad (9)$$

where $\gamma \in (0, 1)$. Then we can compute a stochastic gradient estimator by

$$\mathbf{m}_t = -\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))) \quad (10)$$
$$+ \frac{p_{i,t}}{2|\mathcal{B}_i|}(\nabla g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) + \nabla g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{A}', \mathcal{B}_i)).$$

where $p_{i,t} = \frac{\tau}{\varepsilon'/|\mathcal{S}_i| + u_{i,t-1}} = \nabla f(u_{i,t-1})$. Finally, we can update the model parameter $\mathbf{w}_{t+1}$ by using a momentum-style update or an Adam-style update. The detailed steps are summarized in Algorithm 1, which is referred as SogCLR to emphasize that we aim to optimize the global contrastive objective.

We note that the memory cost of SogCLR is $O(n + d)$, which is $O(d)$ for over-parameterized deep neural networks with $d \gg n$. The per-iteration complexity of SogCLR is the

same as SimCLR.

Next, we provide a convergence result for SogCLR.

Theorem 2. *Assume that* $\mathbb{E}_{\mathcal{A},\mathcal{A}'}\mathbb{E}_{\mathbf{z}\sim\mathcal{S}_i}|E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}) - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})|^2 \leq \epsilon^2$ *for any* $\mathbf{x}_i \in \mathcal{D}$ *and the same conditions as in Theorem 1 hold, then with* $\gamma \leq \frac{n}{B}$, *and* $\eta = O(\min\left\{\beta, \frac{\gamma B}{n}, \frac{1}{L_F}\right\})$, *after* T *iterations, SogCLR ensures that* $\mathbb{E}[\|\nabla F(\mathbf{w}_{t'})\|^2] \leq O(\frac{1}{\eta T} + \frac{\beta+\gamma}{B} + \epsilon^2)$ *for a random* $t' \in \{1, \ldots, T\}$.

Remark: The above theorem implies that by setting $\beta = \sqrt{B/T} < 1$ and $\gamma = \sqrt{n/T} < 1$, then Sog-CLR's optimization error will converge to the level of ϵ when $T = O(\max(\frac{n}{B^2\epsilon^4}, \frac{1}{B\epsilon^4}))$, i.e., $\mathbb{E}[\|\nabla F(\mathbf{w}_{t'})\|^2] \leq O(\frac{1}{\sqrt{BT}} + \frac{\sqrt{n}}{B\sqrt{T}} + \epsilon^2) \leq O(\epsilon^2)$. When ϵ is small enough, the optimization error of SogCLR is negligible. In addition, the analysis also implies that SogCLR enjoys a parallel speed-up, i.e., with a larger mini-batch size B it needs a less number of iterations to converge to a small error.

One might notice that there are two differences between Sog-CLR and the update (7) for SimCLR. One difference is that SogCLR maintains and updates the $\mathbf{u}$ sequence. The second difference is that SogCLR uses a momentum-style update. We would like to emphasize that the moving average update for $\mathbf{u}_{i,t+1}$ is the key to prove the above result. With this technique, SogCLR is able to leverage the momentum-style update or the Adam-style update to enjoy a small optimization error. Without using the scalars $\mathbf{u}_{i,t+1}$ in computing the gradient estimator, even we use the momentum-style update or the Adam-style update for SimCLR, it still suffers from an optimization error in the order of $O(1/\sqrt{B})$. In particular, we have the following corollary for the optimization error of SimCLR with the momentum-style update.

Corollary 1. *Let us consider the following momentum-style update for SimCLR.*
$$\mathbf{v}_t = (1-\beta)\mathbf{v}_{t-1} + \beta\frac{\tau}{B}\sum_{\mathbf{x}_i\in\mathcal{B}}\widehat{\nabla}L(\mathbf{x}_i, \mathcal{A}, \mathcal{A}') \quad (11)$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta\mathbf{v}_t. \quad (12)$$
Assume F *is smooth,* g *is smooth and Lipchitiz continuous, with* $\eta \leq O(\beta)$ *SimCLR ensures that* $\mathbb{E}[\|\nabla F(\mathbf{w}_{t'})\|^2] \leq O(\frac{1}{\eta T} + \frac{1}{\beta T} + \frac{\beta}{B} + \frac{1}{B})$ *for a random* $t' \in \{1, \ldots, T\}$.

Remark: The dominating term in the upper bound is still $O(1/B)$ when $T \to \infty$ and $\beta = O(1/\sqrt{T})$.

3.4. SogCLR optimizes V2 Global Contrastive Objective

In this section, we propose another version of the global contrasive objective (V2) and show that SogCLR optimizes the V2 global contrastive objective, which further justifies the proposed algorithm SogCLR. In particular, let us consider the following objective.

$$F_{v2}(\mathbf{w}) = \mathbb{E}_{\mathbf{x}_i\sim\mathcal{D},\mathcal{A},\mathcal{A}'\sim\mathcal{P}}(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))) \quad (13)$$

$$+ \frac{1}{n}\sum_{\mathbf{x}_i\in\mathcal{D}}\ln\left(\frac{\varepsilon'}{|\mathcal{S}_i|} + \frac{\tau}{|\mathcal{S}_i|}\mathbb{E}_{\mathcal{A}}g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\right) + \text{Const.}$$

The difference between V2 GCO (13) and V1 GCO (5) is that the expectation over $\mathcal{A}$ in the second component is moved from the outside of the logarithmic function to the inside. The above objective function can be also explained from the average of individual contrastive loss. To this end, we define the following contrastive loss for each augmented pair $(\mathcal{A}(\mathbf{x}_i), \mathcal{A}'(\mathbf{x}_i))$:

$$L_2(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{A}') = -\ln\frac{\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))/\tau)}{\varepsilon' + \mathbb{E}_{\mathcal{A}}g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)}.$$
$$(14)$$

Then we have

$$F_{v2}(\mathbf{w}) = \mathbb{E}_{\mathbf{x}\sim\mathcal{D},\mathcal{A},\mathcal{A}'}[\tau L_2(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{A}')].$$
Different from $L(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{A}')$, in the definition of $L_2(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{A}')$ the similarity score of a positive pair $(\mathcal{A}(\mathbf{x}_i), \mathcal{A}'(\mathbf{x}_i))$ is contrasted with all possible negative pairs between $\mathbf{x}_i$ and other images.

We prove that SogCLR indeed converges to a stationary solution to the V2 GCO $F_{v2}(\mathbf{w})$. Different from $F(\mathbf{w})$ defined in (5), the update of u of SogCLR can be considered directly as an moving average estimator of $\mathbb{E}_{\mathcal{A}}g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)$ in $F_{v2}(\mathbf{w})$, which does not involve the error caused by difference between different augmented data. We state the convergence below.

Theorem 3. *Assume the same conditions as in Theorem 1 hold, then with* $\gamma \leq \frac{n}{B}$, *and* $\eta = O(\min\left\{\beta, \frac{\gamma B}{n}, \frac{1}{L_F}\right\})$, *after* T *iterations, SogCLR ensures that* $\mathbb{E}[\|\nabla F_{v2}(\mathbf{w}_{t'})\|^2] \leq O(\frac{1}{\eta T} + \frac{\beta+\gamma}{B})$ *for a random* $t' \in \{1, \ldots, T\}$.

Remark: The above theorem implies that by setting $\beta = \sqrt{B/T} < 1$ and $\gamma = \sqrt{n/T} < 1$, then SogCLR converges to a stationary solution of $F_{v2}(\mathbf{w})$ when $T \to \infty$.

4. Extensions

In this section, we propose the extension of the proposed technique for optimizing other contrastive losses. We note that the large batch size requirement also exists in other contrastive learning methods. Below we consider one task, namely a self-supervised bimodal contrastive learning task.

Optimizing Two-way Contrastive Objective. A recent paper (Radford et al., 2021) proposes a bimodal contrastive learning method named CLIP, which uses a two-way contrastive loss to learn both the encoder network for the image and the encoder network for the text. Radford et al. (2021) uses a very large batch size 32, 768 on a self-collected large-scale dataset with 400 million image and text pairs. Inspired by the SogCLR method for optimizing one-way contrastive

loss and its promising performance, below we present a similar solution to alleviate the requirement of large batch size for optimizing two-way contrastive loss. Given a set of image-text pairs $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{t}_1), \ldots, (\mathbf{x}_n, \mathbf{t}_n)\}$. We denote by E_I and E_T the encoder network for the image data and the text data, respectively. We can consider optimizing a global two-way contrastive loss:

$$F(\mathbf{w}) = -\frac{\tau}{n} \sum_{i=1}^{n} \log \frac{\exp(E_I(\mathbf{x}_i)^\top E_T(\mathbf{t}_i)/\tau)}{\sum_{\mathbf{t} \in \mathcal{D}} \exp(E_I(\mathbf{x}_i)^\top E_T(\mathbf{t})/\tau)}$$
$$-\frac{\tau}{n} \sum_{i=1}^{n} \log \frac{\exp(E_I(\mathbf{x}_i)^\top E_T(\mathbf{t}_i)/\tau)}{\sum_{\mathbf{x} \in \mathcal{D}} \exp(E_I(\mathbf{x})^\top E_T(\mathbf{t}_i)/\tau)}.$$

Due to the large size of $\mathcal{D}$, the challenge lies that handling $g(\mathbf{w}; \mathbf{x}_i) = \mathbb{E}_{\mathbf{t} \sim \mathcal{D}} \exp(E_I(\mathbf{x}_i)^\top E_T(\mathbf{t})/\tau))$ and $g(\mathbf{w}; \mathbf{t}_i) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} \exp(E_I(\mathbf{x})^\top E_T(\mathbf{t}_i)/\tau))$. We propose to compute a stochastic gradient estimator by

$$\mathbf{m}_t = -\frac{1}{B} \sum_{i \in \mathcal{B}} E_I(\mathbf{x}_i)^\top E_T(\mathbf{t}_i) +$$
$$\frac{1}{B} \sum_{i \in \mathcal{B}} \left(\frac{\tau}{u_{i,t}^I} \nabla g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{B}) + \frac{\tau}{u_{i,t}^T} \nabla g(\mathbf{w}_t; \mathbf{t}_i, \mathcal{B}) \right)$$

where $g(\mathbf{w}; \mathbf{x}_i, \mathcal{B})$ and $g(\mathbf{w}; \mathbf{t}_i, \mathcal{B})$ are the mini-batch estimators of $g(\mathbf{w}; \mathbf{x}_i)$ and $g(\mathbf{w}; \mathbf{t}_i)$ respectively. The scalar $u_{i,t+1}^I$ and $u_{i,t+1}^T$ are updated for the sampled data according to

$$u_{i,t+1}^I = (1 - \gamma)u_{i,t}^I + \gamma g(\mathbf{w}_t; \mathbf{x}_i, \mathcal{B})$$
$$u_{i,t+1}^T = (1 - \gamma)u_{i,t}^T + \gamma g(\mathbf{w}_t; \mathbf{t}_i, \mathcal{B}).$$

Then we can update the model by Adam-style update or momentum-style update.

5. Experiments

In this section, we compare SogCLR to SimCLR to demonstrate the effectiveness of our optimization method. For a fair comparison, we adopt the same settings as SimCLR to SogCLR unless noted (the main difference is the batch size). It is not our focus to leverage multiple techniques for achieving state-of-the-art performance (Tomasev et al., 2022). We also compare with the CLIP framework for bimodal contrastive learning. We aim to demonstrate SogCLR can achieve competitive performance when using a smaller batch size. For SimCLR, we run experiments on two scales of ImageNet dataset. The small version is a subset with randomly selected 100 classes (about 128k images) from ImageNet denoted as ImageNet-S (Wu et al., 2019), and the full version of ImageNet (about 1.2 million images) is denoted as ImageNet-1K (Deng et al., 2009). For CLIP, we manually construct a text-image pair dataset based on ImageNet-S using the label of each image to construct a text. For the implementations, we follow these open-source repositories (Chen et al., 2020a; Radford et al., 2021; Ilharco et al., 2021) available on Github. All experiments related to Sim-CLR are trained on Google Cloud TPUs using 8 to 512 cores

depending on model size and batch size. All experiments related to CLIP are trained a NVIDIA V100 GPU with 32GB memory size. The code for SogCLR is available at https://github.com/optimization-ai/sogclr.

5.1. Image Pretraining

Experiment setup. Following previous works (Chen et al., 2020a;b), we pretrain ResNet-50 (He et al., 2016) with a 2-layer 128×128 projection head on top of backbone encoder. We explore different batch sizes of 128, 256, 512 and different training epochs of 100, 200, 400, 800. We use square root learning rate scaling ($0.075 \times$ sqrt(BatchSize)) with a cosine decay schedule without restart. We also use learning rate warm-up for 10 epochs, i.e., learning rate is gradually increased to the maximum value. We follow the same image augmentation strategies as in SimCLR (Chen et al., 2020a;b) including random crop, color distortion, and Gaussian blur. We use LARS optimizer (You et al., 2017) (with a momentum of 0.9 and weight decay of 1e-6) and set temperature(τ) to 0.1 by default for all pretraining experiments. For SogCLR in Algorithm 1, we tune γ in [0.99, 0.9, 0.8, 0.7, 0.6] and initialize sequence $\mathbf{u}_0$ by all zeros. For evaluations, we report performance for linear classifier trained on top of the pretrained encoder on ImageNet validation sets known as linear evaluation (Chen et al., 2020a; He et al., 2020; Caron et al., 2020; Grill et al., 2020). In particular, we train a linear classier using SGD with Nesterov momentum with a batch size of 4096 and learning rate of 0.1 for 90 epochs. For training, We random crop and resize input images to 224×224. For testing, we apply center crop on input images.

Results. We report top-1 accuracy by linear evaluation on ImageNet-S and ImageNet-1K under different batch sizes and training epochs in Table 2 and Table 3. We can see that SogCLR performs consistently better than SimCLR under all settings on two datasets. SogCLR achieves 3.9%, 2.3%, 1.9% average improvements on ImageNet-S and achieves 2.8%, 2.3%, 1.3% average improvements on ImageNet-1K with batch size of 128, 256, 512, respectively. In particular, we achieve 69.4% top-1 accuracy using batch size of 256, which is better than original SimCLR's large-batch (e.g., 4096, 8192) results at 69.1% under the same number of epochs. In addition, we compare the convergence speed of SogCLR with SimCLR using the same batch size of 256 with different number of epochs on ImageNet-S and ImageNet-1K as shown in Figure 4. The results indicate that our algorithm converges faster in terms of number of epochs using small batch sizes.

5.2. Vision and Language Pretraining

Experiment Setup. In this section, we aim to demonstrate our algorithm can also be applied to solve bi-modal self-
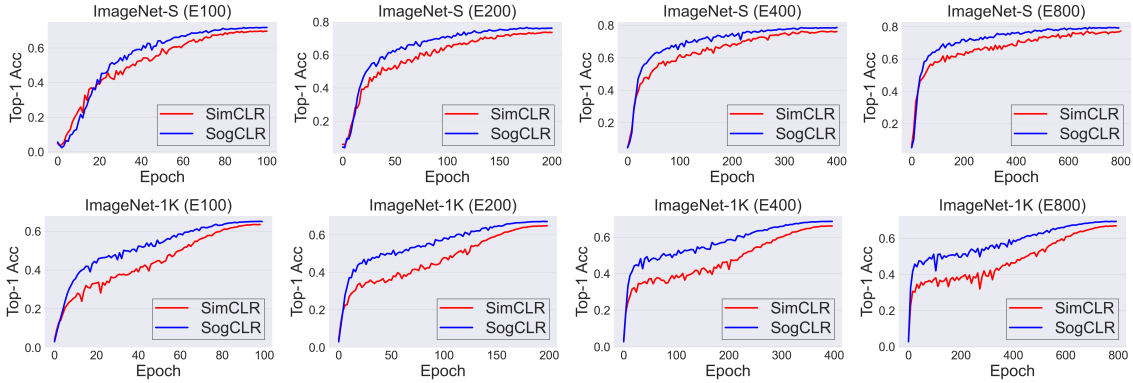
Figure 1. Learning curve for top-1 accuracy by linear evaluation on ImageNet-S and ImageNet-1K trained on R50 with batch size of 256.

Table 2. Linear evaluation (top-1 accuracy) under different batch sizes and training epoch on ResNet-50 and ImageNet-S.

Method	BatchSize\Epoch	100	200	400	800
SimCLR	128	68.5	72.7	75.7	75.7
SogCLR	128	**72.2**	**76.7**	**79.3**	**80.1**
SimCLR	256	69.7	73.6	76.1	77.4
SogCLR	256	**71.8**	**76.3**	**78.7**	**79.4**
SimCLR	512	70.9	74.1	75.9	76.3
SogCLR	512	**71.8**	**75.8**	**78.2**	**79.4**

Table 3. Linear evaluation (top-1 accuracy) under different batch sizes and training epoch on ResNet-50 and ImageNet-1K.

Method	BatchSize\Epoch	100	200	400	800
SimCLR	128	62.6	64.0	64.1	64.5
SogCLR	128	**64.9**	**66.2**	**67.4**	**67.9**
SimCLR	256	62.8	64.3	65.7	66.5
SogCLR	256	**65.2**	**67.1**	**68.7**	**69.4**
SimCLR	512	63.8	65.6	66.7	67.4
SogCLR	512	**65.0**	**67.2**	**68.8**	**69.6**

supervised problems. We study a popular vision and language pretraining framework, i.e., CLIP (Radford et al., 2021). CLIP consists of two parts: vision encoder (e.g., CNN, transformer) and text encoder (e.g., transformer). The original CLIP is pretrained on a large dataset with 400 million image-text pairs to achieve competitive performance against supervised baseline. Here, we are not aiming to achieve the best performance but to study and understand the limits of this framework. Thus, we use the modified CLIP consisting of a modified ResNet-50 and a small vision transformer(ViT) (Dosovitskiy et al., 2021), denoted as CLIP-S. The detailed configuration can be found in Appendix. We use template *"This is a photo of [CLASS]"* to generate the text caption for each image based on ImageNet-S. For training, we use batch size of 128 and 256 to train the models for 30 and 60 epochs. We use warm-up strategy for 1000 iterations to increase learning rate to the maximum value of 0.001 and then decrease it by a cosine decay scheduler. We use Adam-W optimizer (Loshchilov & Hutter, 2017) with the weight decay of 0.1. We set temperature to a fixed value for 0.07 for SogCLR and CLIP. Similar to

Table 4. Top-1 linear evaluation under different batch sizes for bimodal learning on ImageNet-S.

Method	BatchSize\Epoch	30	60
CLIP-S (InfoNCE)	128	67.7	63.4
CLIP-S (SogCLR)	128	**69.5**	**71.3**
CLIP-S (InfoNCE)	256	69.0	64.9
CLIP-S (SogCLR)	256	**69.4**	**70.1**

SimCLR, we tune $\gamma = [0.6 \sim 0.99]$ and set $\mathbf{u}_0$ to zeros for SogCLR. For evaluations, we perform zero-shot evaluation on ImageNet-S validation set using the ensemble results of 80 different prompt templates (Radford et al., 2021). The validation results are presented in Table 4.

Results. We report zero-shot evaluation accuracy of CLIP-S in Table 4. The results indicate that CLIP-S trained by SogCLR performs better than CLIP-S trained by standard InfoNCE loss. In addition, we observe that InfoNCE suffers from 4% performance drop for training 60 epochs. In contrast, SogCLR has a much more stable performance for longer training and achieves over 1% improvement on zero-shot evaluation accuracy. In addition, we also find that CLIP with SogCLR is much more robust to the change of batch size while CLIP with InfoNCE drops more than 1% when switching batch size from 256 to 128.

5.3. Ablation Studies

Verification of algorithmic design and theory. We validate (i) using the momentum update for u_{t+1} (i.e., $\gamma < 1$) is better than without using momentum update ($\gamma = 1$). (ii) $\mathbb{E}_{\mathcal{A},\mathcal{A}',\mathbf{z}}|E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}) - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})|^2 \leq \epsilon^2$ in Theorem 2 holds with a small ϵ^2. In other words, we expect the similarity between the representations of different augmented samples are close. For (i), we train ResNet-50 with batch size of 256 for 100, 200, 400, 800 epochs. We tune the γ in [0.6, 0.7, 0.8, 0.9, 0.99]. The results are summarized in the Table 8 in Appendix. The results indicate that models with $\gamma = 0.7 \sim 0.8$ achieve the best performance. For (ii), we use the models trained with batch size 256 at the checkpoints of 100th, 200th, 400th, 800th epoch to compute $\mathbb{E}_{\mathcal{A},\mathcal{A}',\mathbf{z}}[|\mathcal{A}(\mathbf{x}_i)^\top \mathbf{z} - \mathcal{A}'(\mathbf{x}_i)^\top \mathbf{z}|^2]$ on ImageNet-S, where
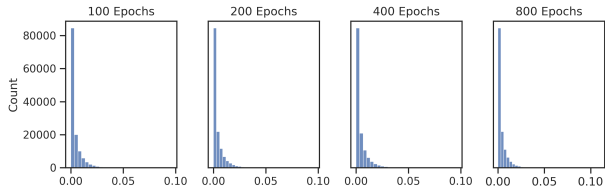
Figure 2. Histogram for difference of learned features between different augmented samples. X-axis is $\mathbb{E}_{\mathcal{A},\mathcal{A}',\mathbf{z}}|E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}) - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})|^2$ and Y-axis denotes the count number.

Table 5. Performance with ResNet-50 $(2\times, 4\times)$ on ImageNet-1K.

Method	Encoder	Params	Batch	Top1	Top5
SimCLR	R50 $(2\times)$	94M	4096	74.2	92.0
SogCLR	R50 $(2\times)$	94M	512	**74.6**	92.1
SimCLR	R50 $(4\times)$	375M	4096	76.5	93.2
SogCLR	R50 $(4\times)$	375M	512	**76.7**	93.1

the expectation is approximated by the Monte Carlo method. We show the histograms of this quantity for all images $\mathbf{x}_i$ in Figure 2, which suggests that all data samples satisfy the above condition in Theorem 2 for some small ϵ.

Impact of batch size. Since SimCLR suffers from the performance drop due to small batch sizes. Here, we show SogCLR is robust to smaller batch sizes. To verify this hypothesis, we train SogCLR using batch sizes varying from 128 to 8192 with ResNet-50 on ImageNet-1K. We set a fixed $\gamma = 0.8$. We directly compare the results taken from Table B.1 in (Chen et al., 2020a) using the same settings. As shown in Figure 3, the performance of SimCLR drops quickly as the decrease of batch size. As a comparison, SogCLR remains stable with batch sizes from 8192 to 256 and there is a small drop for batch size of 128. Overall, SogCLR demonstrates the robustness to different batch sizes. This result is consistent with our theory.
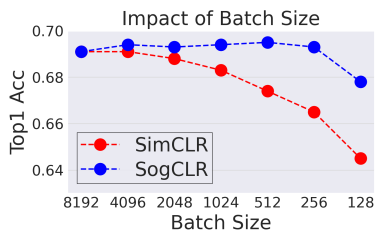

Figure 3. Impact of batch size. Y-axis is linear evaluation accuracy with 800-epoch pretraining with ResNet-50 on ImageNet-1K.

Different network encoders. To verify the effectiveness of the proposed method, we further evaluate it on different network encoders. To this end, we train ResNet models by varying widths. We train ResNet-50 $(2\times, 4\times)$ using batch size of 512 for 800 epochs. We set $\gamma = 0.8$. For baselines, we use the batch size of 4096 to train models for a total of 1000 epochs. The results are summarized in the table below. When using ResNet-50 $(2\times, 4\times)$, we are able to achieve 74.6% and 76.7% top-1 linear evaluation accuracy, which are better than SimCLR's results trained with a larger batch

Table 6. Comparison of small-batch training approaches.

Method	Batch Size\Epochs	100	200	400	800
SimCLR	256	69.7	73.6	76.1	77.4
FlatNCE	256	71.5	75.5	76.7	77.8
SiMo	256	71.5	75.0	76.8	78.2
SogCLR	256	**71.9**	**76.3**	**78.7**	**79.4**

size of 4096 and a large epoch number.

5.4. Comparison with Small Batch Size Methods

We conduct experiments to compare SogCLR with other two InfoNCE-based small-batch training methods, e.g., Flat-NCE (Chen et al., 2021b) and SiMo (Zhu et al., 2020a). We train ResNet-50 with a 2-layer nonlinear projection head on ImageNet-S using a batch size of 256 for 100, 200, 400, 800 epochs. For SiMo, we set $\alpha = 65536$. The results are summarized in Table 6. We observe that all methods outperform SimCLR for 100 and 200 epochs and the improvements for FlatNCE and SiMo seem to disappear when reaching 800 epochs. In contrast, SogCLR performs consistently better.

5.5. Combining with Other Useful Tricks

We explore two commonly used techniques to boost the performance in our framework, namely, multi-layer projection head (Chen et al., 2020a) and multi-crop augmentation (Caron et al., 2020), and incorporate them into SogCLR. We use a 4-layer MLP projection head with 128 neurons for each layer on the top of ResNet-50 encoder, and use a multi-crop strategy by using 4 crops of size 160×160 and 2 crops of size 96×96. SogCLR achieves 72.5% top-1 linear evaluation accuracy with these two tricks and a batch size of 256 and 800 epochs, which is reported in Table 1. We also experiment with different combinations of projection heads and multi-crop data augmentations summarized in Table 9.

6. Conclusion

In this paper, we have examined the large batch size issue in the training of contrastive self-supervised learning from an optimization perspective. To address this issue, we have proposed a global contrastive objective and an efficient stochastic algorithm with provable convergence guarantee. Our analysis also exhibits why existing methods such as SimCLR require a large batch size for ensuring the optimization error to be small. For future work, we plan to incorporate more advanced techniques into the proposed method to further improve the performance.

Acknowledgements

References

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

Chen, J., Gan, Z., Li, X., Guo, Q., Chen, L., Gao, S., Chung, T., Xu, Y., Zeng, B., Lu, W., et al. Simpler, faster, stronger: Breaking the log-k curse on contrastive learners with flatnce. *arXiv preprint arXiv:2107.01152*, 2021a.

Chen, J., Gan, Z., Li, X., Guo, Q., Chen, L., Gao, S., Chung, T., Xu, Y., Zeng, B., Lu, W., et al. Simpler, faster, stronger: Breaking the log-k curse on contrastive learners with flatnce. *arXiv preprint arXiv:2107.01152*, 2021b.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Chen, X. and He, K. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pp. 15750–15758. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Chen_Exploring_Simple_Siamese_Representation_Learning_CVPR_2021_paper.html.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Chen, X., Xie, S., and He, K. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021c.

Chopra, S., Hadsell, R., and LeCun, Y. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 539–546. IEEE, 2005.

Chuang, C.-Y., Robinson, J., Lin, Y.-C., Torralba, A., and Jegelka, S. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Dang, C. D. and Lan, G. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM Journal on Optimization*, 25(2):856–881, 2015.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2021.

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., and Zisserman, A. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9588–9597, 2021.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Guo, Z., Xu, Y., Yin, W., Jin, R., and Yang, T. On stochastic moving-average estimators for non-convex optimization. *arXiv preprint arXiv:2104.14840*, 2021.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pp. 1735–1742, USA, 2006. IEEE Computer Society. ISBN 0769525970. doi: 10.1109/CVPR.2006.100. URL https://doi.org/10.1109/CVPR.2006.100.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Ilharco, G., Wortsman, M., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., Hajishirzi, H., Farhadi, A., and Schmidt, L. Openclip, July 2021. URL https://doi.org/10.5281/zenodo.5143773. If you use this software, please cite it as below.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Li, C., Yang, J., Zhang, P., Gao, M., Xiao, B., Dai, X., Yuan, L., and Gao, J. Efficient self-supervised vision transformers for representation learning. *arXiv preprint arXiv:2106.09785*, 2021.

Li, J., Zhou, P., Xiong, C., and Hoi, S. C. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Mitrovic, J., McWilliams, B., Walker, J., Buesing, L., and Blundell, C. Representation learning via invariant causal mechanisms. *arXiv preprint arXiv:2010.07922*, 2020.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Qi, Q., Luo, Y., Xu, Z., Ji, S., and Yang, T. Stochastic optimization of areas under precision-recall curves with provable convergence. *Advances in Neural Information Processing Systems*, 34, 2021.

Qian, R., Meng, T., Gong, B., Yang, M.-H., Wang, H., Belongie, S. J., and Cui, Y. Spatiotemporal contrastive video representation learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6960–6970, 2021.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems*, pp. 1857–1865, 2016.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33:6827–6839, 2020.

Tomasev, N., Bica, I., McWilliams, B., Buesing, L., Pascanu, R., Blundell, C., and Mitrovic, J. Pushing the limits of self-supervised resnets: Can we outperform supervised learning without labels on imagenet? *arXiv preprint arXiv:2201.05119*, 2022.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., and Fu, Y. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 374–382, 2019.

You, Y., Gitman, I., and Ginsburg, B. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12310–12320. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/zbontar21a.html.

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. *ArXiv*, abs/2106.04560, 2021.

Zhang, Y., Jiang, H., Miura, Y., Manning, C. D., and Langlotz, C. Contrastive learning of medical visual representations from paired images and text, 2021. URL https://openreview.net/forum?id=T4gXBOXoIUr.

Zhu, B., Huang, J., Li, Z., Zhang, X., and Sun, J. Eqco: Equivalent rules for self-supervised contrastive learning. *arXiv preprint arXiv:2010.01929*, 2020a.

Zhu, B., Huang, J., Li, Z., Zhang, X., and Sun, J. Eqco: Equivalent rules for self-supervised contrastive learning. *arXiv preprint arXiv:2010.01929*, 2020b.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., and Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020c.
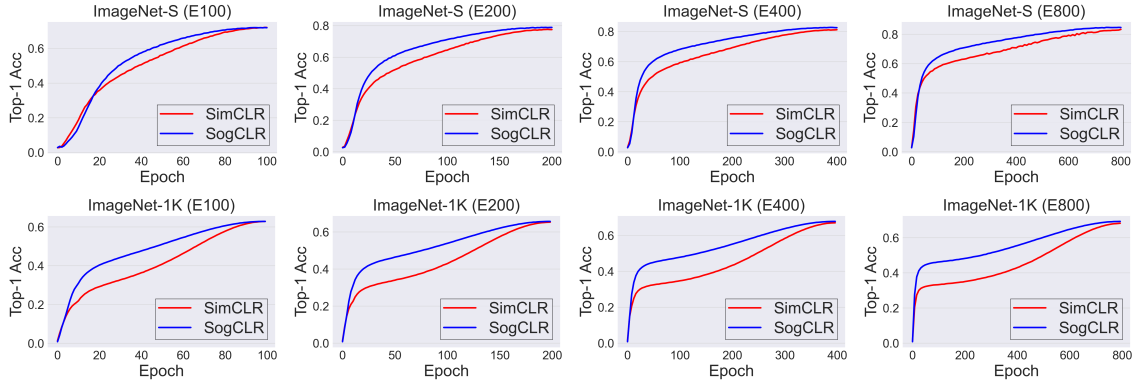
A. Experiment Details

Figure 4. Learning curve for top-1 accuracy by linear evaluation on ImageNet-S and ImageNet-1K **training set** trained on ResNet-50 using batch size of 256.

Table 7. CLIP-S hyper-parameters.

Hyperparameter	Value
embed_dim	512
image_resolution	224×224
vision_layers	[3,4,6,3]
vision_width	32
vision_patch_size	null
context_length	77
vocab_size	49408
transformer_width	128
transformer_heads	8
transformer_layers	8

Table 8. Top-1 linear evaluation accuracy trained on ResNet-50 under different number of epochs using batch size of 256 on ImageNet-1K for $\gamma = 1$ v.s. $\gamma < 1$ in Algorithm 1.

γ\Epoch	100	200	400	800
1.0	62.8	64.3	65.7	66.5
0.99	64.9	67.1	68.3	69.2
0.9	65.0	66.9	68.1	69.2
0.8	**65.2**	67.1	68.4	69.3
0.7	65.0	**67.1**	**68.7**	**69.4**
0.6	64.4	66.7	68.3	69.2

Table 9. Linear evaluations with different nonlinear heads and multi-crop augmentation.

Num of views	3-layer proj. head	4-layer proj. head
2×224	70.7	71.3
$4 \times 160 + 2 \times 96$	71.7	**72.5**

B. Notations in the Proofs

In the following proofs, we abuse the notation: $g_i(\mathbf{w}; \mathcal{A}, \mathcal{S}_i) = g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) = \frac{1}{|\mathcal{S}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$ and $g_i(\mathbf{w}; \mathcal{A}, \mathcal{B}_i) = g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) = \frac{1}{|\mathcal{B}_i|} g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)$. In the following analysis, we assume $\mathbf{x}_i \in \mathcal{B}$ is independently sampled with replacement and $\mathcal{A}, \mathcal{A}'$ are also independently sampled for each sampled data independently. It is notable that

$\mathbb{E}_{\mathcal{B}_i|\mathbf{x}_i}[g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)] = g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)$. We write the objective function as

$$F(\mathbf{w}) = F_1(\mathbf{w}) + F_2(\mathbf{w})$$

where we ignore the constant and

$$F_1(\mathbf{w}) = -\mathbb{E}_{\mathbf{x}_i, \mathcal{A}, \mathcal{A}'}[E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathcal{A}'(\mathbf{x}_i))]$$

$$F_2(\mathbf{w}) = \frac{\tau}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \ln(\varepsilon_0 + g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) = \frac{\tau}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))$$

where $f(g) = \ln(\varepsilon_0 + g)$.

C. Proof of Theorem 1

The SimCLR with the update (7) uses the following gradient estimator:

$$\mathbf{v}_t = \nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) \tag{15}$$

We make the following standard assumptions.

Assumption 1. *We assume that there exist $\sigma, C_g, C_f, L_f, L_F$ such that*

- $\mathbb{E}[\|\nabla F_1(\mathbf{w}; \mathcal{B}) - \nabla F_1(\mathbf{w})\|^2] \le \frac{\sigma^2}{B}$

- $\mathbb{E}_{\mathcal{B}_i|\mathbf{x}_i}[\|g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) - g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)\|^2] \le \frac{\sigma^2}{B}$ *and* $\mathbb{E}_{\mathcal{B}_i|\mathbf{x}_i}[\|\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) - \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)\|^2] \le \frac{\sigma^2}{B}$

- $\|\nabla g_i(\mathbf{w}; \mathcal{A}, \mathcal{S}_i)\| \le C_g$

- $\|\nabla f(g)\| \le C_f$, *and* $\nabla f(\cdot)$ *is L_f Lipschitz continuous*

- F *is L_F-smooth.*

It is notable that the above assumptions are mild or standard for convergence analysis.

Below, we use $\mathbb{E}_t$ to denote the expectation over randomness at t-th iteration given history. First, we have

$$\mathbb{E}_t[F(\mathbf{w}_{t+1})] \le \mathbb{E}_t[F(\mathbf{w}_t) + (\mathbf{w}_{t+1} - \mathbf{w}_t)^\top \nabla F(\mathbf{w}_t) + \frac{\eta^2 L_F}{2} \|\mathbf{v}_t\|^2]$$

$$= F(\mathbf{w}_t) - \eta \mathbb{E}_t[(\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)))]^\top \nabla F(\mathbf{w}_t)$$

$$+ \frac{\eta^2 L_F}{2} \|\mathbf{v}_t\|^2]$$

$$= F(\mathbf{w}_t) - \eta \mathbb{E}_t[(\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)))]^\top \nabla F(\mathbf{w}_t)]$$

$$+ \eta \mathbb{E}_t[(\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)))$$

$$- (\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)))^\top \nabla F(\mathbf{w}_t)$$

$$+ \frac{\eta^2 L_F}{2} \|\mathbf{v}_t\|^2]$$

$$= F(\mathbf{w}_t) - \eta \|\nabla F(\mathbf{w}_t)\|^2 + \eta \mathbb{E}_t[\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \|\nabla F(\mathbf{w}_t)\| C_g L_f \|g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) - g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)\| + \frac{\eta^2 L_F}{2} \|\mathbf{v}_t\|^2]$$

$$= F(\mathbf{w}_t) - \eta \|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta}{2} \|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta C_g^2 L_f^2}{2} \mathbb{E}_t[\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \|g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) - g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)\|^2] + \frac{\eta^2 L_F}{2} \mathbb{E}_t[\|\mathbf{v}_t\|^2]$$

$$= F(\mathbf{w}_t) - \frac{\eta}{2} \|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta C_g^2 L_f^2 \sigma^2}{2B} + \frac{\eta^2 L_F}{2} \mathbb{E}_t[\|\mathbf{v}_t\|^2]$$

Then we have

$$\mathbb{E}[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2] = \mathbb{E}[\|\nabla F_1(\mathbf{w}_t; \mathcal{B}) - \nabla F_1(\mathbf{w}_t)$$

$$+ \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$\leq 2\mathbb{E}[\|\nabla F_1(\mathbf{w}_t; \mathcal{B}) - \nabla F_1(\mathbf{w}_t)\|^2]$$

$$+ 2\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$\leq \frac{2\sigma^2}{B} + 2\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

To bound the second term, we have

$$\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$= \mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i)) - \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))$$

$$+ \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$\leq \mathbb{E}[\frac{2}{B^2} B \sum_{\mathbf{x}_i \in \mathcal{B}} C_g^2 L_f^2 \|g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) - g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)\|^2]$$

$$+ 2\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$= \frac{2C_g^2 L_f^2 \sigma^2}{B}$$

$$+ 2\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{B}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) - \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} [\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))$$

$$+ \frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} [\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$= \frac{2C_g^2 L_f^2 \sigma^2}{B} + \frac{4C_f^2 \sigma^2}{B}$$

$$+ 4\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} [\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i)) - \frac{1}{n} \sum_{\mathbf{x}_i \in \mathcal{D}} \mathbb{E}_\mathcal{A} \nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$\leq \frac{2C_g^2 L_f^2 \sigma^2}{B} + \frac{4C_f^2 \sigma^2}{B}$$

$$+ 4\mathbb{E}[\|\frac{1}{B} \sum_{\mathbf{x}_i \in \mathcal{B}} [\nabla g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i) \nabla f(g(\mathbf{w}; \mathbf{x}_i, \mathcal{A}, \mathcal{S}_i))\|^2]$$

$$\leq \frac{2C_g^2 L_f^2 \sigma^2}{B} + \frac{4C_f^2 \sigma^2}{B} + 4C_g^2 C_f^2$$

As a result,

$$\mathbb{E}[\|\mathbf{v}_t\|^2] \leq 2\|\nabla F(\mathbf{w}_t)\|^2 + \frac{C}{B} + 16C_g^2 C_f^2,$$

where C is a proper constant. By combining the above results together, we have

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq F(\mathbf{w}_t) - \frac{\eta}{2}\|\nabla F(\mathbf{w}_t)\|^2 + \frac{\eta C}{B} + \eta^2 L_F \|\nabla F(\mathbf{w}_t)\|^2 + 16\eta^2 L_F C_g^2 C_f^2.$$

Then with $\eta L_F \leq 1/4$, we have

$$\mathbb{E}[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F(\mathbf{w}_t)\|^2] \leq \frac{4(F(\mathbf{w}_1)-F_*)}{\eta T} + 64\eta L_f C_f^2 C_g^2 + \frac{4C}{B},$$

which finises the proof.

We can also sharpen the bound of $\mathbb{E}[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2]$ by noting that

$$\mathbb{E}[\|\frac{1}{B}\sum_{\mathbf{x}_i \in \mathcal{B}}[\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)) - \frac{1}{n}\sum_{\mathbf{x}_i \in \mathcal{D}}\mathbb{E}_{\mathcal{A}}\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i))\|^2]$$

$$= \mathbb{E}[\|\frac{1}{B}\sum_{\mathbf{x}_i \in \mathcal{B}}[\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)) - \frac{1}{B}\sum_{\mathbf{x}_i \in \mathcal{B}}\mathbb{E}_{\mathcal{A}}[\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i))+$$

$$\frac{1}{B}\sum_{\mathbf{x}_i \in \mathcal{B}}\mathbb{E}_{\mathcal{A}}[\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)) - \frac{1}{n}\sum_{\mathbf{x}_i \in \mathcal{D}}\mathbb{E}_{\mathcal{A}}\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\nabla f(g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i))\|^2]$$

$$\leq \frac{4C_g^2 C_f^2}{B}.$$

As a result, $\mathbb{E}[\|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2] \leq \frac{2C_g^2 L_f^2 \sigma^2}{B} + \frac{4C_f^2 \sigma^2}{B} + \frac{4C_g^2 C_f^2}{B}$, then with $\eta L_F \leq 1/4$, we have

$$\mathbb{E}[\frac{1}{T}\sum_{t=1}^{T}\|\nabla F(\mathbf{w}_t)\|^2] \leq \frac{4(F(\mathbf{w}_1)-F_*)}{\eta T} + \frac{C}{B},$$

which still has a dependence of $1/B$. However, we can set $\eta = O(1)$ and $T = O(1/\epsilon^2), B = O(1/\epsilon^2)$ in order to achieve an ϵ-stationary solution.

D. Proof of Theorem 2

First, we note that the gradient estimator $\mathbf{m}_t$ is

$$\mathbf{m}_t = \nabla F_1(\mathbf{w}_t;\mathcal{B}) + \frac{1}{B}\sum_{\mathbf{x}_i \in \mathcal{B}}\nabla f(u_{i,t})\underbrace{\frac{1}{2}(\nabla g(\mathbf{w}_t;\mathbf{x}_i,\mathcal{A},\mathcal{B}_i) + \nabla g(\mathbf{w}_t;\mathbf{x}_i,\mathcal{A}',\mathcal{B}_i))}_{\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)}$$

$$\mathbf{u}_{i,t+1} = (1-\gamma)\mathbf{u}_{i,t} + \gamma\underbrace{\frac{1}{2}(g(\mathbf{w}_t;\mathbf{x}_i,\mathcal{A},\mathcal{B}_i) + g(\mathbf{w}_t;\mathbf{x}_i,\mathcal{A}',\mathcal{B}_i))}_{g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)}$$

Define $g_i(\mathbf{w}) = \mathbb{E}_{\mathcal{A}}[g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)]$. We can see that $\mathbb{E}_{\mathcal{B}_i,\mathcal{A},\mathcal{A}'|\mathbf{x}_i}[g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)] = g_i(\mathbf{w}_t)$, and $\mathbb{E}_{\mathcal{B}_i,\mathcal{A},\mathcal{A}'|\mathbf{x}_i}[\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)] = \nabla g_i(\mathbf{w}_t)$.

We make the following assumptions.

Assumption 2. *We assume that there exist $\sigma, C_g, C_f, L_f, L_F$ such that*

- $\mathbb{E}[\|\nabla F_1(\mathbf{w};\mathcal{B}) - \nabla F_1(\mathbf{w})\|^2] \leq \frac{\sigma^2}{B}$

- $\mathbb{E}_{\mathcal{B}_i|\mathbf{x}_i}[\|g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{B}_i) - g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\|^2] \leq \frac{\sigma^2}{B}$ *and* $\mathbb{E}_{\mathcal{B}_i|\mathbf{x}_i}[\|\nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{B}_i) - \nabla g(\mathbf{w};\mathbf{x}_i,\mathcal{A},\mathcal{S}_i)\|^2] \leq \frac{\sigma^2}{B}$

- $\|\nabla f(g)\| \leq C_f, \|\nabla g_i(\mathbf{w})\| \leq C_g, \|\nabla F_1(\mathbf{w})\| \leq C_{F_1}, \|\nabla g_i(\mathbf{w};\mathcal{A},\mathcal{S}_i)\| \leq C_g$

- $\nabla f(\cdot)$ *is L_f Lipschitz continuous*

- F *is L_F-smooth.*

- $\|E(\mathbf{z})\| \leq 1, \forall \mathbf{z}$

- $\mathbb{E}_{\mathcal{A},\mathcal{A}'}\mathbb{E}_{\mathbf{z} \sim \mathcal{S}_i}|E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z}) - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})|^2 \leq \epsilon^2$ *for any $\mathbf{x}_i \in \mathcal{D}$*

We note that under the above assumption we have

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{A},\mathcal{A}'}\|g_i(\mathbf{w}_t;\mathcal{A}',\mathcal{S}_i) - g_i(\mathbf{w}_t;\mathcal{A},\mathcal{S}_i)\|^2 \\
&= \mathbb{E}_{\mathcal{A},\mathcal{A}'}\|\mathbb{E}_{\mathbf{z}\sim\mathcal{S}_i}(\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z})/\tau) - \mathbb{E}_{\mathbf{z}\sim\mathcal{S}_i}(\exp(E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})/\tau)\|^2 \\
&\leq \mathbb{E}_{\mathcal{A},\mathcal{A}'}\mathbb{E}_{\mathbf{z}\sim\mathcal{S}_i}\|(\exp(E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z})/\tau) - (\exp(E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})/\tau)\|^2 \\
&\leq C\mathbb{E}_{\mathcal{A},\mathcal{A}'}\mathbb{E}_{\mathbf{z}\sim\mathcal{S}_i}\|E(\mathcal{A}(\mathbf{x}_i))^\top E(\mathbf{z})/\tau - E(\mathcal{A}'(\mathbf{x}_i))^\top E(\mathbf{z})/\tau\|^2 \\
&\leq O(\epsilon^2)
\end{aligned}
$$

where C is a proper constant that bounds the Lipschitz of $\exp(E(\dot{})E(\cdot)/\tau)$.

We need the following lemma, whose proof can be found in (Guo et al., 2021) and thus is omitted here.

Lemma 1. *Consider a sequence $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta\mathbf{v}_t$ and the L_F-smooth function F and the step size $\eta_t L_F \leq 1/2$.*

$$
F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \frac{\eta}{2}\|\Delta_t\|^2 - \frac{\eta}{2}\|\nabla F(\mathbf{w}_t)\|^2 - \frac{\eta}{4}\|\mathbf{v}_t\|^2, \tag{16}
$$

where $\Delta_t = \mathbf{v}_t - \nabla F(\mathbf{w}_t)$.

Lemma 2. *Assume $\mathbb{E}_{\mathcal{A},\mathcal{A}'}\|g_i(\mathbf{w}_t;\mathcal{A}',\mathcal{S}_i) - g_i(\mathbf{w}_t;\mathcal{A},\mathcal{S}_i)\|^2 \leq \epsilon^2$, we have*

$$
\mathbb{E}[\|\Delta_t\|^2] \leq (1-\beta)\mathbb{E}\left[\|\Delta_{t-1}\|^2\right] + \frac{2L_F^2\eta^2\mathbb{E}\left[\|\mathbf{v}_{t-1}\|^2\right]}{\beta} + \frac{14\beta L_f^2 C_g^2}{n}\mathbb{E}\left[\|\Xi_t\|^2\right] + \frac{14\beta L_f^2 C_g^2}{n}\mathbb{E}\left[\|\mathbf{u}_t - \mathbf{u}_{t+1}\|^2\right]
$$

$$
+ \frac{\beta^2 C}{B} + 5\beta C_g^2 L_f^2\epsilon^2.
$$

where $\Delta_t = \mathbf{v}_t - \nabla F(\mathbf{w}_t)$, $\Xi_t = \mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_t)$, $\mathbf{g}(\mathbf{w}) = (g_1(\mathbf{w}),\ldots,g_n(\mathbf{w}))$, and C is a proper constant.

Proof. We define that $\Delta_t = \mathbf{v}_t - \nabla F(\mathbf{w}_t)$. Below, for the analysis of Δ_t, we note that $\mathbf{u}_t$ is independent of the randomness in $\mathcal{B},\mathcal{A},\mathcal{A}'$. Based on the update rule $\mathbf{v}_t = (1-\beta)\mathbf{v}_{t-1} + \beta\mathbf{m}_t$, we have

$$
\|\Delta_t\|^2 = \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2
$$

$$
= \left\|(1-\beta)\mathbf{v}_{t-1} + \beta(\nabla F_1(\mathbf{w}_t;\mathcal{B}) + \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f([\mathbf{u}_{t-1}]_i)\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)) - \nabla F(\mathbf{w}_t)\right\|^2
$$

$$
= \left\|\underbrace{(1-\beta)(\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1}))}_{A_1} + \underbrace{(1-\beta)(\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t))}_{A_2}\right.
$$

$$
+ \beta\underbrace{\left(\frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t))\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i) - \frac{1}{B}\sum_{i\in\mathcal{B}}\frac{1}{2}\sum_{\mathbf{a}=\mathcal{A},\mathcal{A}'}\nabla f(g_i(\mathbf{w}_t;\mathbf{a},\mathcal{S}_i))\nabla g_i(\mathbf{w}_t;\mathbf{a},\mathcal{B}_i)\right)}_{A_3}
$$

$$
+ \beta\underbrace{\left(\frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f([\mathbf{u}_{t-1}]_i)\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i) - \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t))\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)\right)}_{A_4}
$$

$$
+ \beta\underbrace{\left(\nabla F_1(\mathbf{w}_t;\mathcal{B}) + \frac{1}{2B}\sum_{i\in\mathcal{B}}(\nabla f(g_i(\mathbf{w}_t;\mathcal{A},\mathcal{S}_i))\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{B}_i) + \nabla f(g_i(\mathbf{w}_t;\mathcal{A}',\mathcal{S}_i))\nabla g_i(\mathbf{w}_t;\mathcal{A}',\mathcal{B}_i)) - \nabla F(\mathbf{w}_t)\right)}_{A_5}\left.\right\|^2.
$$

Note that $\mathbb{E}\left[\langle A_1, A_5\rangle\right] = \mathbb{E}\left[\langle A_2, A_5\rangle\right] = 0$. Then,

$$
\mathbb{E}_t\left[\|A_1 + A_2 + A_3 + A_4 + A_5\|^2\right] = \|A_1\|^2 + \|A_2\|^2 + \mathbb{E}_t\left[\|A_3\|^2\right] + \mathbb{E}_t\left[\|A_4\|^2\right] + \mathbb{E}_t\left[\|A_5\|^2\right] + 2\langle A_1, A_2\rangle
$$

$$
+ 2\mathbb{E}_t\left[\langle A_1, A_3\rangle\right] + 2\mathbb{E}_t\left[\langle A_1, A_4\rangle\right] + 2\mathbb{E}_t\left[\langle A_2, A_3\rangle\right] + 2\mathbb{E}_t\left[\langle A_2, A_4\rangle\right] + 2\mathbb{E}_t\left[\langle A_3, A_4\rangle\right] + 2\mathbb{E}_t\left[\langle A_3, A_5\rangle\right] + 2\mathbb{E}_t\left[\langle A_4, A_5\rangle\right].
$$

Based on Young's inequality for products, we have $2 \langle \mathbf{a}, \mathbf{b} \rangle \leq \frac{\|\mathbf{a}\|^2 c}{2} + \frac{2\|\mathbf{b}\|^2}{c}$ for $c > 0$.

$$\mathbb{E}_t \left[\|A_1 + A_2 + A_3 + A_4 + A_5\|^2 \right]$$

$$\leq (1 + \beta) \|A_1\|^2 + (3 + 3/\beta) \|A_2\|^2 + (4 + 3/\beta)\mathbb{E}_t \left[\|A_3\|^2 \right] + (4 + \frac{3}{\beta})\mathbb{E}_t \left[\|A_4\|^2 \right] + 3\mathbb{E}_t \left[\|A_5\|^2 \right].$$

Thus, we have

$$\mathbb{E}_t[\|\Delta_t\|^2] \leq (1 - \beta) \|\Delta_{t-1}\|^2 + (3 + 3/\beta) \|A_2\|^2 + (4 + 3/\beta)\mathbb{E}_t[\|A_3\|^2] + (4 + 3/\beta)\mathbb{E}_t[\|A_4\|^2] + 3\mathbb{E}_t \left[\|A_5\|^2 \right]. \tag{17}$$

Moreover, we have

$$\|A_2\|^2 = (1 - \beta)^2 \|\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t)\|^2 \leq (1 - \beta)^2 \eta^2 L_F^2 \|\mathbf{v}_{t-1}\|^2, \tag{18}$$

$$\mathbb{E}_t[\|A_3\|^2] \leq \beta^2 C_g^2 L_f^2 \mathbb{E}_{\mathcal{A}} \|g_i(\mathbf{w}_t) - g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)\|^2 \leq \beta^2 C_g^2 L_f^2 \mathbb{E}_{\mathcal{A}} \|\mathbb{E}_{\mathcal{A}'} g_i(\mathbf{w}_t; \mathcal{A}', \mathcal{S}_i) - g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)\|^2$$

$$\leq \beta^2 C_g^2 L_f^2 \mathbb{E}_{\mathcal{A},\mathcal{A}'} \|g_i(\mathbf{w}_t; \mathcal{A}', \mathcal{S}_i) - g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)\|^2 \tag{19}$$

$$\mathbb{E}_t[\|A_4\|^2] \leq \mathbb{E}_t[\frac{\beta^2}{B} \sum_{i \in \mathcal{B}} \|\nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{A}', \mathcal{B}_i)\|^2 \|\nabla f([\mathbf{u}_{t-1}]_i) - \nabla f(g_i(\mathbf{w}_t))\|^2] \tag{20}$$

$$\leq \mathbb{E}_t[\frac{\beta^2 L_f^2}{B} \sum_{i \in \mathcal{B}} \|\nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{A}', \mathcal{B}_i)\|^2 \|[\mathbf{u}_{t-1}]_i - g_i(\mathbf{w}_t)\|^2]$$

$$\leq \beta^2 L_f^2 C_g^2 \mathbb{E}_t \left[\frac{1}{B} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t-1}]_i - g_i(\mathbf{w}_t)\|^2 \right] \tag{21}$$

Since the update rule of $\mathbf{u}_t$ is based on $\mathcal{B}$, we have

$$\mathbb{E}_t \left[\frac{1}{B} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t-1}]_i - g_i(\mathbf{w}_t)\|^2 \right] = \frac{1}{n} \mathbb{E}_t \left[\|\mathbf{u}_{t-1} - \mathbf{g}(\mathbf{w}_t)\|^2 \right],$$

where and $\mathbf{g}(\mathbf{w}_t) := [g_1(\mathbf{w}_t), \cdots, g_n(\mathbf{w}_t)]^\top$. Then, $\|\mathbf{u}_{t-1} - \mathbf{g}(\mathbf{w}_t)\|^2 = \sum_{i \in \mathcal{D}} \|[\mathbf{u}_{t-1}]_i - g_i(\mathbf{w}_t)\|^2$. Thus, we also have

$$\mathbb{E}_t \left[\|A_5\|^2 \right] \leq \frac{2\beta^2 \sigma^2}{B}$$

$$+ 2\beta^2 \mathbb{E}_t \left[\left\| \left(\frac{1}{2B} \sum_{i \in \mathcal{B}} (\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i) + \nabla f(g_i(\mathbf{w}_t; \mathcal{A}', \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}', \mathcal{B}_i)) - \nabla F_2(\mathbf{w}_t) \right) \right\|^2 \right]$$
$$\tag{22}$$

$$\leq \frac{2\beta^2 \sigma^2}{B} + 4\beta^2 \mathbb{E}_t \left[\frac{1}{B} \sum_{i \in \mathcal{B}} \mathbb{E}_{\mathcal{B}_i} \|(\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_{t+1}; \mathcal{A}, \mathcal{B}_i) - \mathbb{E}_{\mathcal{B}_i} \nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i))\|^2 \right]$$

$$+ 8\beta^2 \mathbb{E}_{\mathcal{B}} \left[\frac{1}{B} \sum_{i \in \mathcal{B}} \mathbb{E}_{\mathcal{A}} \|\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i) - \mathbb{E}_{\mathcal{A}} \nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)\|^2 \right]$$

$$+ 8\beta^2 \mathbb{E}_{\mathcal{B}} \left[\left\| \frac{1}{B} \sum_{i \in \mathcal{B}} \mathbb{E}_{\mathcal{A}} \nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i) - \frac{1}{n} \sum_{i \in \mathcal{D}} \mathbb{E}_{\mathcal{A}} \nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i)) \nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i) \right\|^2 \right]$$

$$\leq \frac{\beta^2 C}{B},$$

where C is some proper constant.

Define that $\Xi_t = \mathbb{E} \left[\frac{1}{n} \|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_t)\|^2 \right]$. By combining the above inequalities we have

$$\mathbb{E}[\|\Delta_t\|^2] \leq (1 - \beta)\mathbb{E} \left[\|\Delta_{t-1}\|^2 \right] + \frac{3L_F^2 \eta^2 \mathbb{E} \left[\|\mathbf{v}_{t-1}\|^2 \right]}{\beta} + \frac{14\beta L_f^2 C_g^2}{n} \mathbb{E} \left[\|\Xi_t\|^2 \right] + 14\beta L_f^2 C_g^2 \mathbb{E} \left[\|\mathbf{u}_t - \mathbf{u}_{t-1}\|^2 \right]$$

$$+ \frac{\beta^2 C}{B} + 7\beta C_g^2 L_f^2 \epsilon^2.$$

$\square$

Lemma 3. *If $\gamma \le 1/5$, the following equation holds.*

$$\mathbb{E}\left[\Xi_{t+1}\right] \le \left(1 - \frac{\gamma B}{4n}\right)\mathbb{E}\left[\Xi_t\right] + \frac{5n\eta^2 C_g^2 \mathbb{E}\left[\|\mathbf{v}_t\|^2\right]}{\gamma B} + \frac{2\gamma^2\sigma^2 B}{nB} - \frac{1}{4n}\mathbb{E}\left[\|\mathbf{u}_{t+1} - \mathbf{u}_t\|^2\right]. \tag{23}$$

By Combining Lemma 1, 2, and Lemma 3, we can prove the final theorem.

$$\mathbb{E}\left[F(\mathbf{w}_{t+1}) - F^*\right] \le \mathbb{E}\left[F(\mathbf{w}_t) - F^*\right] + \frac{\eta}{2}\mathbb{E}\left[\Delta_t\right] - \frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] - \frac{\eta}{4}\mathbb{E}\left[\|\mathbf{v}_t\|^2\right] \tag{24}$$

$$\mathbb{E}\left[\Delta_{t+1}\right] \le (1-\beta)\mathbb{E}\left[\Delta_t\right] + \frac{3L_F^2\eta^2}{\beta}\mathbb{E}\left[\|\mathbf{v}_t\|^2\right] + 14\beta L_f^2 C_g^2 \mathbb{E}\left[\Xi_{t+1}\right] + \frac{\beta^2 C}{B} + \frac{14\beta L_f^2 C_g^2}{n}\mathbb{E}\left[\|\mathbf{u}_{t+1} - \mathbf{u}_t\|^2\right]$$
$$+ 7\beta C_g^2 L_f^2 \epsilon^2 \tag{25}$$

$$\mathbb{E}\left[\Xi_{t+1}\right] \le \left(1 - \frac{\gamma B}{4n}\right)\mathbb{E}\left[\Xi_t\right] + \frac{5n\eta^2 C_g^2 \mathbb{E}\left[\|\mathbf{v}_t\|^2\right]}{\gamma B} + \frac{2\gamma^2\sigma^2 B}{nB} - \frac{1}{4n}\mathbb{E}\left[\|\mathbf{u}_{t+1} - \mathbf{u}_t\|^2\right]. \tag{26}$$

Summing (24), $\frac{\eta}{\beta}\times$(25), and $\frac{56L_f^2 C_g^2 n\eta}{\gamma B}\times$(26) leads to

$$\mathbb{E}\left[(F(\mathbf{w}_{t+1}) - F^*) + \frac{\eta}{\beta}\Delta_{t+1} + \frac{56L_f^2 C_g^2 n\eta}{\gamma B}\left(1 - \frac{\gamma B}{4n}\right)\Xi_{t+1}\right]$$

$$\le \mathbb{E}\left[(F(\mathbf{w}_t) - F^*) + \frac{\eta}{\beta}\left(1 - \frac{\beta}{2}\right)\Delta_t + \frac{56L_f^2 C_g^2 n\eta}{\gamma B}\left(1 - \frac{\gamma B}{4n}\right)\Xi_t\right] - L_f^2 C_g^2 \eta \left(\frac{14n}{\gamma B} - 14\right)\mathbb{E}\left[\frac{1}{n}\|\mathbf{u}_{t+1} - \mathbf{u}_t\|^2\right]$$

$$- \frac{\eta}{2}\mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] - \eta\left(\frac{1}{4} - \frac{3L_F^2\eta^2}{\beta^2} - \frac{280L_f^2 n^2 C_g^4 \eta^2}{\gamma^2 B^2}\right)\mathbb{E}\left[\|\mathbf{v}_t\|^2\right] + \frac{\beta\eta C}{B} + \frac{112\eta\gamma L_f^2 C_g^2 \sigma^2}{B} + 7\eta C_g^2 L_f^2 \epsilon^2.$$

If $\gamma \le \frac{n}{B}$, we have $\frac{14n}{\gamma B} - 14 \ge 0$. Set $\beta = O(\min(B\epsilon^2, \frac{2}{7}))$, $\gamma = \min\left\{O(B\epsilon^2), \frac{5n}{14B}\right\}$, and $\eta = \min\left\{\frac{\beta}{6L_F}, \frac{\gamma B}{50L_f n C_g^2}\right\}$.
Define the Lyapunov function as $\Phi_t := (F(\mathbf{w}_t) - F^*) + \frac{\eta}{\beta}\Delta_t + \frac{56L_f^2 C_g^2}{B}\frac{\eta}{\gamma}\left(1 - \frac{\gamma B}{4n}\right)\Xi_t$. If we initialize $\mathbf{v}_0 = 0$, we have
$\mathbb{E}[\Delta_1] \le 2C_{F_1} + 2C_f^2 C_g^2$. Then,

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] \le \frac{2\Lambda_\Phi^1}{\eta T} + \frac{2\beta C}{B} + \frac{224\gamma L_f^2 C_g^2 \sigma^2}{B} + 14C_g^2 L_f^2 \epsilon^2, \tag{27}$$

where we define $\mathbb{E}\left[\Phi_1\right] \le \Delta_F + \frac{1}{6L_F}(2C_{F_1}^2 + 2C_f^2 C_g^2) + \frac{C\Xi_1}{n} =: \Lambda_\Phi^1$. After $T = O(\max(\frac{n}{B^2\epsilon^4}, \frac{1}{B\epsilon^4}))$ iterations, we have
$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] \le O(\epsilon^2)$.

E. Proof of Corollary 1

The proof is similar to that of Theorem 2, except that the bound Δ_t, which is shown below.

$$\|\Delta_t\|^2 = \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2$$

$$= \left\| (1-\beta)\mathbf{v}_{t-1} + \beta(\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i))\nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i)) - \nabla F(\mathbf{w}_t) \right\|^2$$

$$= \left\| \underbrace{(1-\beta)(\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1}))}_{A_1} + \underbrace{(1-\beta)(\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t))}_{A_2} \right.$$

$$+ \beta \underbrace{\left(\frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i))\nabla g_i(\mathbf{w}_{t+1}; \mathcal{A}, \mathcal{B}_i) - \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i))\nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i) \right)}_{A_3}$$

$$+ \beta \underbrace{\left. \left(\nabla F_1(\mathbf{w}_t; \mathcal{B}) + \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{S}_i))\nabla g_i(\mathbf{w}_t; \mathcal{A}, \mathcal{B}_i) - \nabla F(\mathbf{w}_t) \right) \right\|^2}_{A_4} .$$

Note that $\mathbb{E}\left[\langle A_1, A_4 \rangle\right] = \mathbb{E}\left[\langle A_2, A_4 \rangle\right] = 0$. Then,

$$\mathbb{E}_t\left[\|A_1 + A_2 + A_3 + A_4\|^2\right] = \|A_1\|^2 + \|A_2\|^2 + \mathbb{E}_t\left[\|A_3\|^2\right] + \mathbb{E}_t\left[\|A_4\|^2\right] + 2\langle A_1, A_2 \rangle$$

$$+ 2\mathbb{E}_t\left[\langle A_1, A_3 \rangle\right] + 2\mathbb{E}_t\left[\langle A_1, A_4 \rangle\right] + 2\mathbb{E}_t\left[\langle A_2, A_3 \rangle\right] + 2\mathbb{E}_t\left[\langle A_2, A_4 \rangle\right] + 2\mathbb{E}_t\left[\langle A_3, A_4 \rangle\right]$$

Based on Young's inequality for products, we have $2\langle \mathbf{a}, \mathbf{b} \rangle \leq \frac{\|\mathbf{a}\|^2 c}{2} + \frac{2\|\mathbf{b}\|^2}{c}$ for $c > 0$.

$$\mathbb{E}_t\left[\|A_1 + A_2 + A_3 + A_4 + A_5\|^2\right]$$

$$\leq (1+\beta)\|A_1\|^2 + C/\beta\|A_2\|^2 + C/\beta\mathbb{E}_t\left[\|A_3\|^2\right] + C\mathbb{E}_t\left[\|A_4\|^2\right],$$

where C is a proper constant. We can show that $\mathbb{E}[\|A_3\|^2] \leq \beta^2\frac{C}{B}$ and $\mathbb{E}[\|A_4\|^2] \leq \beta^2\frac{C}{B}$ for some constant C. Then we have

$$\mathbb{E}[\|\Delta_t\|^2] \leq (1-\beta)\mathbb{E}\left[\|\Delta_{t-1}\|^2\right] + \frac{C\eta^2\mathbb{E}\left[\|\mathbf{v}_{t-1}\|^2\right]}{\beta} + \frac{\beta C}{B} + \frac{\beta^2 C}{B}$$

Combining this inequality with lemma 1 and with $\eta \leq O(\beta)$, we can prove an optimization error

$$\mathbb{E}[\frac{1}{T}\sum_t \|\nabla F(\mathbf{w}_t)\|^2] \leq O(\frac{1}{\eta T} + \frac{1}{\beta T} + \frac{\beta}{B} + \frac{1}{\beta})$$

F. Proof of Theorem 3

The proof is similar to that of Theorem 2, except that the bound Δ_t, which is shown below.

$$\|\Delta_t\|^2 = \|\mathbf{v}_t - \nabla F(\mathbf{w}_t)\|^2$$

$$= \left\| (1-\beta)\mathbf{v}_{t-1} + \beta\left(\nabla F_1(\mathbf{w}_t;\mathcal{B}) + \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f([\mathbf{u}_t]_i)\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)\right) - \nabla F(\mathbf{w}_t) \right\|^2$$

$$= \left\| \underbrace{(1-\beta)(\mathbf{v}_{t-1} - \nabla F(\mathbf{w}_{t-1}))}_{A_1} + \underbrace{(1-\beta)(\nabla F(\mathbf{w}_{t-1}) - \nabla F(\mathbf{w}_t))}_{A_2} \right.$$

$$+ \underbrace{\beta\left(\frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f([\mathbf{u}_t]_i)\nabla g_i(\mathbf{w}_{t+1};\mathcal{A},\mathcal{A}',\mathcal{B}_i) - \frac{1}{B}\sum_{i\in\mathcal{B}}\nabla f(g_i(\mathbf{w}_t))\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{A}',\mathcal{B}_i)\right)}_{A_4}$$

$$+ \left. \underbrace{\beta\left(\nabla F_1(\mathbf{w}_t;\mathcal{B}) + \frac{1}{2B}\sum_{i\in\mathcal{B}}(\nabla f(g_i(\mathbf{w}_t))\nabla g_i(\mathbf{w}_t;\mathcal{A},\mathcal{B}_i) + \nabla f(g_i(\mathbf{w}_t))\nabla g_i(\mathbf{w}_t;\mathcal{A}',\mathcal{B}_i)) - \nabla F(\mathbf{w}_t)\right)}_{A_5} \right\|^2 .$$

from which we can see that A_3 is gone in the proof of Theorem 2, which is the source to cause the error depends on ϵ. Then we can follow the same analysis to finish the proof, which is omitted here due to that it is almost a duplicate of Theorem 2.

F.1. Proof of Lemma 3

Proof. Based on Algorithm, the update rule of $[\mathbf{u}]_i$ is

$$[\mathbf{u}_{t+1}]_i = \begin{cases} (1-\gamma)[\mathbf{u}_t]_i + \gamma g_i(\mathbf{w}_{t+1};\xi) & i\in\mathcal{B} \\ [\mathbf{u}_t]_i & i\notin\mathcal{B} \end{cases}.$$

where ξ denote the randomness in $\mathcal{A},\mathcal{A}',\mathcal{B}_i$, and $\mathbf{w}_{t+1} = \mathbf{w}_t$. Below, we dnote by $B_1 = |\mathcal{B}|$ and $\mathbb{E}[\|g_i(\mathbf{w}_{t+1};\xi) - g_i(\mathbf{w}_{t+1})\|^2] \leq \frac{\sigma^2}{B_2}$, where $B_1 = B_2 = B = |\mathcal{B}|$. We can re-write it into the equivalent expression below.

$$[\mathbf{u}_{t+1}]_i = \begin{cases} [\mathbf{u}_t]_i - \gamma\left([\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1};\xi)\right) & i\in\mathcal{B} \\ [\mathbf{u}_t]_i & i\notin\mathcal{B} \end{cases}. \tag{28}$$

Let us define $\phi_t(\mathbf{u}) = \frac{1}{2}\|\mathbf{u} - \mathbf{g}(\mathbf{w}_t)\|^2 = \frac{1}{2}\sum_{i=1}^n\|\mathbf{u}_i - g_i(\mathbf{w}_t)\|^2$, which is a 1-strongly convex function. Then, the update rule (28) can be viewed as one step of the stochastic block mirror descent algorithm (Algorithm 2 in Dang & Lan (2015)) for minimizing $\phi_{t+1}(\mathbf{u})$, where the Bregman divergence is associated with the quadratic function. We follow the analysis of Dang & Lan (2015).

$$\phi_{t+1}(\mathbf{u}_{t+1}) = \frac{1}{2}\|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_{t+1})\|^2$$

$$= \frac{1}{2}\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 + \langle\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1}), \mathbf{u}_{t+1} - \mathbf{u}_t\rangle + \frac{1}{2}\|\mathbf{u}_{t+1} - \mathbf{u}_t\|^2$$

$$= \frac{1}{2}\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 + \sum_{i\in\mathcal{B}}\langle[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1};\xi), [\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\rangle + \frac{1}{2}\sum_{i\in\mathcal{B}}\|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2$$

$$+ \sum_{i\in\mathcal{B}}\langle g_i(\mathbf{w}_{t+1};\xi) - g_i(\mathbf{w}_{t+1}), [\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\rangle .$$

Note that $[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; B_2) = ([\mathbf{u}_t]_i - [\mathbf{u}_{t+1}]_i)/\gamma$ and $2\langle b - a, a - c\rangle \leq \|b - c\|^2 - \|a - b\|^2 - \|a - c\|^2$.

$$\sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), [\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i \rangle$$

$$= \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle + \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), [\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1}) \rangle$$

$$= \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle + \frac{1}{\gamma} \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - [\mathbf{u}_{t+1}]_i, [\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1}) \rangle$$

$$\leq \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle$$

$$+ \frac{1}{2\gamma} \sum_{i \in \mathcal{B}} \left(\|[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1})\|^2 - \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2 - \|[\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1})\|^2 \right)$$

If $\gamma < \frac{1}{5}$, we have

$$-\frac{1}{2}\left(\frac{1}{\gamma} - 1 - \frac{(\gamma + 1)}{4\gamma}\right) \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2 + \sum_{i \in \mathcal{B}} \langle g_i(\mathbf{w}_{t+1}; \xi) - g_i(\mathbf{w}_{t+1}), [\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i \rangle$$

$$\leq -\frac{1}{4\gamma} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2 + \gamma \sum_{i \in \mathcal{B}} \|g_i(\mathbf{w}_{t+1}; \xi) - g_i(\mathbf{w}_{t+1})\|^2 + \frac{1}{4\gamma} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2$$

$$= \gamma \sum_{i \in \mathcal{B}} \|g_i(\mathbf{w}_{t+1}; \xi) - g_i(\mathbf{w}_{t+1})\|^2.$$

Then, we have

$$\frac{1}{2} \|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_{t+1})\|^2$$

$$\leq \frac{1}{2} \|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 + \frac{1}{2\gamma} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1})\|^2 - \frac{1}{2\gamma} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1})\|^2$$

$$+ \gamma \sum_{i \in \xi} \|g_i(\mathbf{w}_{t+1}; \xi) - g_i(\mathbf{w}_{t+1})\|^2 + \sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle - \frac{(\gamma + 1)}{8\gamma} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2.$$

Note that $\frac{1}{2\gamma} \sum_{i \notin \mathcal{B}} \|[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1})\|^2 - \frac{1}{2\gamma} \sum_{i \notin \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1})\|^2 = 0$ based on Algorithm, which implies that

$$\frac{1}{2\gamma} \sum_{i \in \mathcal{B}} \left(\|[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1})\|^2 - \|[\mathbf{u}_{t+1}]_i - g_i(\mathbf{w}_{t+1})\|^2 \right) = \frac{1}{2\gamma_t} \left(\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_t)\|^2 - \|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_t)\|^2 \right).$$

Besides, we also have:

$$\mathbb{E}_t \left[\sum_{i \in \mathcal{B}} \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}; \xi), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle \right] = \frac{B_1}{n} \sum_{i=1}^n \langle [\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1}), g_i(\mathbf{w}_{t+1}) - [\mathbf{u}_t]_i \rangle$$

$$= -\frac{B_1}{n} \|[\mathbf{u}_t]_i - g_i(\mathbf{w}_{t+1})\|^2$$

$$\mathbb{E} \left[\sum_{i \in \mathcal{B}} \|g_i(\mathbf{w}_{t+1}; \xi) - g_i(\mathbf{w}_{t+1})\|^2 \right] \leq \frac{B_1 \sigma^2}{B_2}.$$

Then, we can obtain

$$\frac{\gamma + 1}{2} \mathbb{E} \left[\|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_{t+1})\|^2 \right] \leq \frac{\gamma \left(1 - \frac{B_1}{n}\right) + 1}{2} \mathbb{E} \left[\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 \right] + \frac{\gamma^2 B_1 \sigma^2}{B_2} - \frac{(\gamma + 1)}{8} \sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2.$$

Further divide $\frac{\gamma + 1}{2}$ and take full expectation on both sides

$$\mathbb{E} \left[\|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_{t+1})\|^2 \right] \leq \frac{\gamma \left(1 - \frac{B_1}{n}\right) + 1}{\gamma + 1} \mathbb{E} \left[\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 \right] + \frac{2}{1 + \gamma} \frac{\gamma^2 \sigma^2 B_1}{B_2} - \frac{1}{4} \mathbb{E} \left[\sum_{i \in \mathcal{B}} \|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2 \right].$$

Note that $\frac{\gamma\left(1 - \frac{B_1}{n}\right) + 1}{\gamma + 1} = 1 - \frac{\gamma B_1}{(\gamma + 1)n} \leq 1 - \frac{\gamma B_1}{2n}$ and $\frac{1}{1+\gamma} \leq 1$ for $\gamma \in (0, 1]$. Besides, we have $\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2 \leq (1 + \frac{\gamma B_1}{4n}) \|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_t)\|^2 + (1 + \frac{4n}{\gamma B_1}) \|\mathbf{g}(\mathbf{w}_{t+1}) - \mathbf{g}(\mathbf{w}_t)\|^2$ due to Young's inequality and $\|\mathbf{g}(\mathbf{w}_t) - \mathbf{g}(\mathbf{w}^{t-1})\|^2 \leq$

$$nC_g^2 \|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2 = n\eta^2 C_g^2 \|\mathbf{v}^t\|^2.$$

$$\mathbb{E}\left[\Xi_{t+1}\right] = \mathbb{E}\left[\frac{1}{n}\|\mathbf{u}_{t+1} - \mathbf{g}(\mathbf{w}_{t+1})\|^2\right]$$

$$\leq \left(1 - \frac{\gamma B_1}{2n}\right)\mathbb{E}\left[\frac{1}{n}\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_{t+1})\|^2\right] + \frac{2\gamma^2\sigma^2 B_1}{nB_2} - \frac{1}{4n}\mathbb{E}\left[\sum_{i\in\mathcal{B}}\|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2\right]$$

$$\leq \left(1 - \frac{\gamma B_1}{2n}\right)\left(1 + \frac{\gamma B_1}{4n}\right)\mathbb{E}\left[\frac{1}{n}\|\mathbf{u}_t - \mathbf{g}(\mathbf{w}_t)\|^2\right] + \frac{5nC_g^2\mathbb{E}\left[\|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2\right]}{\gamma B_1} + \frac{2\gamma^2\sigma^2 B_1}{nB_2}$$

$$- \frac{1}{4n}\mathbb{E}\left[\sum_{i\in\mathcal{B}}\|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2\right]$$

$$\leq \left(1 - \frac{\gamma B_1}{4n}\right)\mathbb{E}\left[\Xi_t\right] + \frac{5n\eta^2 C_g^2\mathbb{E}\left[\|\mathbf{v}^t\|^2\right]}{\gamma B_1} + \frac{2\gamma^2\sigma^2 B_1}{nB_2} - \frac{1}{4n}\mathbb{E}\left[\sum_{i\in\mathcal{B}}\|[\mathbf{u}_{t+1}]_i - [\mathbf{u}_t]_i\|^2\right].$$

$\square$