Stability-Based Generalization Analysis of the Asynchronous Decentralized SGD

Xiaoge Deng, Tao Sun*, Shengwei Li, and Dongsheng Li*

National Lab for Parallel and Distributed Processing (PDL), College of Computer,
National University of Defense Technology, Changsha, Hunan, China.
dengxg@nudt.edu.cn, nudtsuntao@163.com, lucasleesw9@gmail.com, dsli@nudt.edu.cn

Abstract

The generalization ability often determines the success of machine learning algorithms in practice. Therefore, it is of great theoretical and practical importance to understand and bound the generalization error of machine learning algorithms. In this paper, we provide the first generalization results of the popular stochastic gradient descent (SGD) algorithm in the distributed asynchronous decentralized setting. Our analysis is based on the uniform stability tool, where stable means that the learned model does not change much in small variations of the training set. Under some mild assumptions, we perform a comprehensive generalizability analysis of the asynchronous decentralized SGD, including generalization error and excess generalization error bounds for the strongly convex, convex, and non-convex cases. Our theoretical results reveal the effects of the learning rate, training data size, training iterations, decentralized communication topology, and asynchronous delay on the generalization performance of the asynchronous decentralized SGD. We also study the optimization error regarding the objective function values and investigate how the initial point affects the excess generalization error. Finally, we conduct extensive experiments on MNIST, CIFAR-10, CIFAR-100, and Tiny-ImageNet datasets to validate the theoretical findings.

Introduction

Stochastic gradient descent (SGD) (Robbins and Monro 1951) has become the mainstay for training modern machine learning (ML) models. In practice, the solution founded by SGD has not only very small training error but also generalizes surprisingly well to the test data (Zhang et al. 2021). Recently, a series of theoretical researches have been dedicated to establishing the generalization error bounds for SGD, i.e., the expected difference between empirical risk on finite training data and population risk on unseen test examples. The seminal work (Hardt, Recht, and Singer 2016) tackled this problem with the algorithmic stability of SGD, which measures sensitivity to perturbations in the training dataset. Stability-based generalization analysis is an important learning theory topic that has been further improved and extended in (Kuzborskij and Lampert 2018; Lei and Ying 2020; Bassily et al. 2020; Zhang et al. 2022).

*Corresponding author.

Currently, distributed training becomes ubiquitous as the scale and complexity of ML models and datasets increase significantly (Dean et al. 2012; Li et al. 2014). In distributed SGD (Zinkevich et al. 2010; Li et al. 2014), multiple workers process data in parallel and communicate with the parameter server, which then updates the model accordingly. This implement often suffers from 1) full synchronization overhead from all workers and 2) communication bottleneck of the central server. To deal with these issues, *asynchronous decentralized stochastic gradient descent* (AD-SGD) was proposed in (Lian et al. 2018) to improve the training efficiency of the distributed systems. In AD-SGD, workers do not wait for all others and only communicate in a decentralized manner, thus enabling wait-free computation and communication. The asynchronous decentralized algorithm has been widely studied due to its superior performance (Jiang et al. 2021; Cui et al. 2021; Nadiradze et al. 2021; Lan and Zhou 2021; Xu, Zhang, and Wang 2021).

From the theoretical perspective, although there exists plenty of convergence analysis of AD-SGD (Lian et al. 2018; Nadiradze et al. 2021; Jiang et al. 2021; Xu, Zhang, and Wang 2021), whether the converged solution generalizes well on unseen testing data under the asynchronous decentralized setting has not been explored. This study aims to close the theoretical gap in the generalizability of the AD-SGD algorithm with a stability-based analytical framework.

Our Contributions

This paper studies the generalization behavior of AD-SGD for the first time, including the generalization error and excess generalization error. We give comprehensive theoretical results covering strongly convex, convex, and non-convex problems and further provide a detailed analysis of two commonly used learning rates. The main results are listed in Table 1, and our contributions can be summarized as follows.

- We study the algorithmic stability of AD-SGD and derive its generalization error bounds accordingly. Theoretical results show that the impact of the learning rate, training data size, and training iterations on AD-SGD is identical to the SGD algorithm (Hardt, Recht, and Singer 2016). More importantly, we reveal the joint effects of decentralized communication topology and asynchronous delay on the stability and generalizability of AD-SGD.

Bound	Learning rate	μ-Strongly convex	Convex	Non-convex
ϵ_{gen}	$\alpha_t = \alpha$	$\mathcal{O}\left\{\frac{1}{\mu n} + (\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m})\frac{\alpha}{\mu}\right\}$	$\mathcal{O}\left\{\frac{T\alpha}{n} + (\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m})\frac{T\alpha^2}{m}\right\}$	$\mathcal{O}\left\{\frac{1+n\beta\alpha-\lambda}{n(1-\lambda)}(1 + \frac{\beta\alpha}{m})^T\right\}$
	$\alpha_t = \mathcal{O}(\frac{1}{t+1})$	$\mathcal{O}\left\{\frac{1}{\mu n} + \frac{mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}}}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T}{T}\right\}$	$\mathcal{O}\left\{\frac{\ln T}{n} + (\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m})\right\}$	$\mathcal{O}\left\{\frac{1+nm-\lambda}{n(1-\lambda)}T\right\}$
ϵ_{opt}	$\alpha_t = \alpha$	$\mathcal{O}\left\{\frac{\alpha}{\mu} + (\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m})\frac{r\alpha}{\mu}\right\}$	$\mathcal{O}\left\{(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m})r\alpha + \frac{r^2}{T\alpha}\right\}$	$\mathcal{O}\left\{(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m})\frac{\alpha}{\gamma} + \frac{r}{T\gamma\alpha}\right\}$
	$\alpha_t = \mathcal{O}(\frac{1}{t+1})$	$\mathcal{O}\left\{\frac{r(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T}{T}\right\}$	$\mathcal{O}\left\{(r^2 + \frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{\overline{\tau}}{m})\frac{1}{\ln T}\right\}$	$\mathcal{O}\left\{[r + \frac{1}{\gamma^2}(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{\overline{\tau}}{m})]\frac{1}{\ln T}\right\}$

Table 1: Summary of the generalization error ϵ_{gen} and optimization error ϵ_{opt} bounds. According to decomposition (1), the excess generalization error ϵ_{exc} is upper bounded by the summation of these two terms. The results are obtained after T iterations of AD-SGD with m distributed workers on n training samples, where $\overline{\tau}$ is the maximum asynchronous delay, and λ characterizes the properties of decentralized topology. β, γ, r and C_λ are constants used in the analysis.

- For excess generalization error, we investigate the optimization errors of AD-SGD in terms of the objective function values. Compared to previous work (Sun, Li, and Wang 2021), we analyze the more interesting non-convex case under the Polyak-Łojasiewicz condition. In addition, we give the uniform stability of the ergodic average model for consistency with the optimization error.
- We conduct extensive experiments to validate our theoretical findings. For the general convex case, we test a linear model on the MNIST dataset. We then use the non-convex ResNet-18 and VGG-16 models for experiments on CIFAR-10, CIFAR-100, and Tiny-ImageNet datasets. We explore the generalization performance of the AD-SGD algorithm under different learning rates, communication topologies, and asynchronous delays.

Related Work

Decentralized and asynchronous SGD. Decentralized algorithms can be traced back to (Tsitsiklis 1984), which do not specify any central node for distributed training. Decentralized SGD (D-SGD) was studied in (Nedic and Ozdaglar 2009; Tang et al. 2018; Assran et al. 2019; Sun and Li 2020; Sun, Li, and Wang 2022), which is based on partial averaging and has been shown to outperform its centralized counterparts (Lian et al. 2017). Asynchronous training (Tsitsiklis, Bertsekas, and Athans 1986; Recht et al. 2011) breaks the limitation of synchronization by allowing all clients to work independently and makes the distributed system well tolerant to the straggler problem. Asynchronous SGD (A-SGD) (Agarwal and Duchi 2011; Lian et al. 2015; Zheng et al. 2017) ensures a wait-free update with delayed gradients, and the convergence is guaranteed under bounded asynchrony.

Asynchronous decentralized SGD (AD-SGD) was studied in (Sirb and Ye 2016; Lian et al. 2018). Sirb and Ye provided a convergence analysis of AD-SGD with mild assumptions. Lian et al. further demonstrated that AD-SGD is superior to asynchronous or decentralized SGD and scales well with large-scale distributed training systems. Luo et al. (2020) proposed a new communication primitive as well as simple group generation and scheduling techniques to optimize AD-SGD. Recently, various variants based on the AD-SGD algorithm have been investigated. Nadiradze et al. (2021) consider AD-SGD with quantization and local updates to further reduce communication overhead. Xu, Zhang, and Wang (2021) proposed a private version of AD-SGD to prevent inference by malicious participants. AD-SGD has also been applied to training acoustic models (Cui et al. 2021) and online learning (Jiang et al. 2021).

Stability and generalization of SGD. Fundamental work (Bousquet and Elisseeff 2002) defined notions of algorithmic stability and revealed the connection between stability and generalization error. Subsequent work (Elisseeff et al. 2005) investigated stability bounds for randomized learning algorithms. Data-dependent stability was proposed in (Shalev-Shwartz et al. 2010) and further studied in (Kuzborskij and Lampert 2018; Lei and Ying 2020; Zhou, Liang, and Zhang 2022). The seminal work (Hardt, Recht, and Singer 2016) derived the uniform stability bounds of SGD through the expansion properties of stochastic gradients. Stability-based generalization analysis was also developed for Langevin dynamics (Mou et al. 2018), pairwise learning (Lei, Ledent, and Kloft 2020), and minimax problems (Lei et al. 2021; Xing, Song, and Cheng 2021).

While the above studies do not consider the distributed settings, (Wu, Zhang, and Wang 2019) provided a stability-based generalization analysis for divide-and-conquer distributed learning algorithms. Based on uniform stability, (Regatti et al. 2019) investigated the generalization performance of A-SGD, and (Sun, Li, and Wang 2021) provided stability and generalization bounds for D-SGD. (Zhu et al. 2022) studied the impact of communication topology on the generalizability of D-SGD with an on-average stability tool. However, (Regatti et al. 2019) only explored generalization errors under centralized parameter server architecture. (Sun, Li, and Wang 2021) also does not consider the effect of asynchrony on decentralized SGD and omits the excess generalization bound for the non-convex case. In this study, we provide comprehensive stability and generalizability results of SGD in the composite asynchronous decentralized setting.

Preliminaries

This section contains a priori knowledge about stability and generalization as well as a description of the AD-SGD algorithm. Throughout the paper, we use the following notation.

Notation. For a vector $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|$ represents its ℓ_2-norm. $\mathbb{E}[\cdot]$ denotes the expectation of $\cdot$ with respect to the underly-

ing probability space. $(\cdot)^\top$ denotes the transpose of the corresponding matrix or vector. $\mathbf{1}_m$ and $\mathbf{0}$ are column vectors in $\mathbb{R}^m$ and $\mathbb{R}^d$; all elements are 1 and 0, respectively.

Stability and Generalization

Let $\mathcal{S} = \{\mathbf{z}_1, \cdots, \mathbf{z}_n\}$ be a set of training examples drawn independently and identically from an underlying distribution $\mathcal{D}$. The aim is to learn a good model $\mathbf{x}$ from a parameter space $\Omega \subseteq \mathbb{R}^d$ based on the training dataset $\mathcal{S}$. The loss function $f(\mathbf{x}; \mathbf{z})$ measures the performance of a model $\mathbf{x}$ on an example $\mathbf{z}$. The corresponding population and empirical risks are respectively defined as

$$F(\mathbf{x}) = \mathbb{E}_{\mathbf{z}\sim\mathcal{D}}[f(\mathbf{x}; \mathbf{z})] \quad \text{and} \quad F_{\mathcal{S}}(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^{n} f(\mathbf{x}; \mathbf{z}_j).$$

Directly measuring the population risk can be difficult as the underlying distribution $\mathcal{D}$ is usually unknown. The ML community then considers solving an approximate empirical risk minimization problem. For a specific stochastic algorithm $\mathcal{A}$, $\mathbf{x} = \mathcal{A}(\mathcal{S})$ denotes the output model obtained by minimizing empirical risk on the training dataset $\mathcal{S}$. The *generalization error* is defined as the expected difference between the population risk and the empirical risk

$$\epsilon_{\text{gen}} = \mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F(\mathcal{A}(\mathcal{S})) - F_{\mathcal{S}}(\mathcal{A}(\mathcal{S}))\right].$$

Here the expectation is over the randomness of algorithm $\mathcal{A}$ and training samples $\mathcal{S}$. A popular tool to bound generalization error ϵ_{gen} is to consider the *uniform stability* (Bousquet and Elisseeff 2002; Hardt, Recht, and Singer 2016).

Definition 1 (Uniform stability) *A stochastic algorithm $\mathcal{A}$ is ϵ_{stab}-uniformly stable if for any datasets $\mathcal{S}, \mathcal{S}'$ which differ in at most one example, we have*

$$\sup_{\mathbf{z}} \mathbb{E}_{\mathcal{A}}\left[f(\mathcal{A}(\mathcal{S}); \mathbf{z}) - f(\mathcal{A}(\mathcal{S}'); \mathbf{z})\right] \leq \epsilon_{\text{stab}}.$$

Uniform stability reflects the effect of changing one training example on the model performance, and its relationship with generalization error is established in the following lemma [Theorem 2.2, (Hardt, Recht, and Singer 2016)].

Lemma 1 (Generalization error via uniform stability) *Let the stochastic algorithm $\mathcal{A}$ be ϵ_{stab}-uniformly stable. Then the generalization error satisfies*

$$\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F(\mathcal{A}(\mathcal{S})) - F_{\mathcal{S}}(\mathcal{A}(\mathcal{S}))\right] \leq \epsilon_{\text{stab}}.$$

Since the ultimate goal of ML algorithms is to minimize the population risk $F(\mathcal{A}(\mathcal{S}))$, this study is also interested in the *excess generalization error*

$$\epsilon_{\text{exc}} = \mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F(\mathcal{A}(\mathcal{S})) - F(\mathbf{x}^*)\right],$$

where $\mathbf{x}^*$ is the minimizer of F. Let $\mathbf{x}^*_{\mathcal{S}}$ be the minimizer of $F_{\mathcal{S}}$, and the optimization error ϵ_{opt} is defined as the difference between the empirical risk and minimum empirical risk in expectation. Then, ϵ_{exc} can be decomposed as

$$\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F(\mathcal{A}(\mathcal{S})) - F(\mathbf{x}^*)\right] = \underbrace{\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F(\mathcal{A}(\mathcal{S})) - F_{\mathcal{S}}(\mathcal{A}(\mathcal{S}))\right]}_{\epsilon_{\text{gen}}}$$
$$+ \underbrace{\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F_{\mathcal{S}}(\mathcal{A}(\mathcal{S})) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})\right]}_{\epsilon_{\text{opt}}} + \underbrace{\mathbb{E}_{\mathcal{S},\mathcal{A}}\left[F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}}) - F(\mathbf{x}^*)\right]}_{\leq 0}.$$

The last term is negative due to the unbiased expectation $\mathbb{E}_{\mathcal{S},\mathcal{A}}[F(\mathbf{x}^*)] = \mathbb{E}_{\mathcal{S},\mathcal{A}}[F_{\mathcal{S}}(\mathbf{x}^*)]$ and $F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}}) \leq F_{\mathcal{S}}(\mathbf{x}^*)$. We then mainly focus on the first two parts and have

$$\epsilon_{\text{exc}} \leq \epsilon_{\text{gen}} + \epsilon_{\text{opt}}. \tag{1}$$

Following Lemma 1, the generalization error ϵ_{gen} is controlled by the algorithmic uniform stability. Therefore, it is sufficient to study the stability for bounding ϵ_{gen}. The optimization error ϵ_{opt} is induced by running the optimization algorithm to minimize the empirical risk, which will be addressed with the optimization theory.

Asynchronous Decentralized SGD

Consider a distributed system with m computing workers to train a ML model under data parallelism. There are two popular ways of dividing data samples. One is that each worker has a uniform random sampling from the training dataset $\mathcal{S}$ with replacement. The other is randomly dividing $\mathcal{S}$ into m disjoint subsets with equal cardinality. Our analysis holds for both variants. The empirical risk can be reformulated as

$$F_{\mathcal{S}}(\mathbf{x}) = \frac{1}{nm}\sum_{i=1}^{m}\sum_{j=1}^{n} f(\mathbf{x}; \mathbf{z}_{j(i)}), \tag{2}$$

where $\{\mathbf{z}_{j(i)}\}_{1\leq j\leq n}$ is the training data stored in worker i.

Distributed SGD minimizes the empirical risk (2) as follows: each worker computes stochastic gradient with local data and sends it to a centralized server; the server synchronizes the gradients from all workers and then updates the model in a gradient descent way. The communication complexity of the central node is $\mathcal{O}(m)$, which severely hampers the scalability of the distributed system (Lian et al. 2017). In addition, there is a gradient-averaging synchronization barrier at each iteration, which also reduces the training efficiency and even fails the training (Assran et al. 2020).

AD-SGD was proposed in (Lian et al. 2018), which uses m distributed workers to optimize problem (2) asynchronously according to a decentralized communication graph $\mathcal{G}$. In AD-SGD, the communication complexity of the busiest worker is only $\mathcal{O}(\deg(\mathcal{G}))$, and the idle time due to synchronization is also reduced. $\mathcal{G} = (\mathbf{V}, \mathbf{W})$ is an undirected connected graph, where $\mathbf{V} = \{1, 2, \cdots, m\}$ denotes the set of m computational workers. $\mathbf{W} = [w_{ij}] \in \mathbb{R}^{m\times m}$ is a doubly stochastic matrix, which satisfies

- if worker j is connected to worker i, or $i = j$, then $w_{ij} > 0$; otherwise, $w_{ij} = 0$;
- $\mathbf{W}^\top = \mathbf{W}$, $\mathbf{W}\mathbf{1}_m = \mathbf{1}_m$, and $\mathbf{1}_m^\top\mathbf{W} = \mathbf{1}_m^\top$.

We denote λ_i as the i-th largest eigenvalue of $\mathbf{W}$ and define a crucial constant $\lambda = \max\{|\lambda_2|, |\lambda_m(\mathbf{W})|\}$. From the definition of the doubly stochastic matrix, we have $0 \leq \lambda < 1$. For a connected graph, $\lambda = 0$ implies that the communication topology is fully-connected, i.e., all elements of $\mathbf{W}$ are $\frac{1}{m}$. In the decentralized setting, we only concern $0 < \lambda < 1$.

Let $\mathbf{x}(i) \in \mathbb{R}^d$ be the local model kept in worker i. The AD-SGD algorithm can be described as follows. 1). Each worker i computes the stochastic gradient $\nabla f(\hat{\mathbf{x}}(i); \mathbf{z}_{j(i)})$ with its local data $\mathbf{z}_{j(i)}$, where $\hat{\mathbf{x}}(i)$ is the model read in local

memory; 2). All workers partially average their local models according to the decentralized communication matrix $\mathbf{W}$

$$\mathbf{X} \leftarrow \mathbf{X}\mathbf{W} \text{ where } \mathbf{X} = [\mathbf{x}(1)\ \mathbf{x}(2)\ \cdots\ \mathbf{x}(m)] \in \mathbb{R}^{d\times m};$$

3). Worker i updates the local model with the staled gradient

$$\mathbf{x}(i) \leftarrow \mathbf{x}(i) - \alpha\nabla f(\hat{\mathbf{x}}(i); \mathbf{z}_{j(i)}),$$

where α is the learning rate. Note that the entire training process is performed asynchronously, then $\mathbf{x}(i)$ may differ from $\hat{\mathbf{x}}(i)$ in step 3) as it can be modified in the averaging step. In AD-SGD, each worker runs the procedures above on its own without any global synchronization. We then denote each gradient update as one iteration, and the iterative format of AD-SGD can be viewed as

$$\mathbf{X}_{t+1} = \mathbf{X}_t\mathbf{W} - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t; \mathbf{z}_{j_t}), \tag{3}$$

where α_t is the learning rate applied at the t-th iteration and

$$\mathbf{G}(\hat{\mathbf{X}}_t; \mathbf{z}_{j_t}) = [\mathbf{0}\ \cdots\ \nabla f(\hat{\mathbf{x}}_t(i_t); \mathbf{z}_{j_t(i_t)})\ \cdots\ \mathbf{0}] \in \mathbb{R}^{d\times m}.$$

Here, i_t denotes the worker that performs gradient update in the t-th iteration. $j_t(i_t)$ is the index of the training data selected in the t-th iteration. τ_t records the asynchronous delay after performing t iterations and $\hat{\mathbf{x}}_t(i_t) = \mathbf{x}_{t-\tau_t}(i_t)$. We have simplified the relevant notation without causing ambiguity, the details of which can be found in (Lian et al. 2018). For any i, $\mathbf{x}_t(i)$ is the local model on the i-th worker in the t-th iteration, and we assume that all workers started from the same model. The output of AD-SGD is the consensus model $\mathbf{x} = \sum_{i=1}^{m}\mathbf{x}(i)$, which is the focus of our analysis.

Stability and Generalization Errors

This section presents the generalization error bounds based on the uniform stability of AD-SGD. Our analysis uses the following standard assumptions.

Assumption 1 (Lipschitz) *The loss function $f(\mathbf{x};\mathbf{z}) : \Omega \to \mathbb{R}$ is differentiable with respect to $\mathbf{x}$ and L-Lipschitz for every $\mathbf{z}$, i.e.,*

$$|f(\mathbf{y};\mathbf{z}) - f(\mathbf{x};\mathbf{z})| \le L\|\mathbf{y}-\mathbf{x}\|.$$

Assumption 2 (Smoothness) *The loss function $f(\mathbf{x};\mathbf{z}) : \Omega \to \mathbb{R}$ is β-smooth for every $\mathbf{z}$, i.e.,*

$$\|\nabla f(\mathbf{y};\mathbf{z}) - \nabla f(\mathbf{x};\mathbf{z})\| \le \beta\|\mathbf{y}-\mathbf{x}\|.$$

Assumption 3 (Bounded delay) *The asynchronous delay is bounded, i.e., there exists a constant $\overline{\tau}$ such that $\tau_t \le \overline{\tau}$ for all iteration t.*

Assumption 4 (Bounded space) *The parameter space Ω is bounded by a closed ball in $\mathbb{R}^d$.*

The Lipschitz assumption indicates that the gradient of the loss function is bounded, which is necessary for the analysis of uniform stability (Hardt, Recht, and Singer 2016; Regatti et al. 2019; Sun, Li, and Wang 2021). Assumption 4 is used for bounding the excess generalization error, which is easy to hold with the projection operation (Hardt, Recht, and Singer 2016; Lei and Ying 2020; Sun, Li, and Wang 2021). Asynchronous and decentralized training makes the stability analysis of AD-SGD more complicated than SGD, A-SGD, and D-SGD. We first give two key lemmas to bound the errors introduced by decentralization and asynchrony.

Lemma 2 *Assume that the loss function is L-Lipschitz and all workers started from the same model, i.e., $\mathbf{x}_1(1) = \cdots = \mathbf{x}_1(m)$. Then the difference between the consensus model $\mathbf{x}_t$ and each local model $\mathbf{x}_t(i)$ is bounded, i.e.,*

$$\|\mathbf{x}_t - \mathbf{x}_t(i)\| \le L\sum_{s=1}^{t-1}\alpha_s\lambda^{t-1-s}.$$

Lemma 3 *Assume that the loss function is L-Lipschitz. Then the deviation from asynchronous delayed updates is bounded, i.e.,*

$$\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| \le \frac{L}{m}\sum_{s=t-\tau_t}^{t-1}\alpha_s.$$

Subsequently, we study the uniform stability of AD-SGD in the convex, strongly convex, and non-convex cases and then derive the generalization error bound according to Lemma 1. Due to space limitations, please refer to the supplementary materials for detailed theoretical proofs.

Convex Case

Theorem 1 *Assume that the loss function is convex, L-Lipschitz, and β-smooth. If the learning rate $\alpha_t \le 2m/\beta$, then the uniform stability ϵ_{stab} of AD-SGD after T iterations is bounded by*

$$\sum_{t=1}^{T-1}\Big[\frac{2L^2\alpha_t}{nm} + \frac{2\beta L^2\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big].$$

The first term comes from the different samples between datasets $\mathcal{S}$ and $\mathcal{S}'$, which means that increasing the amount of training data can make the algorithm more stable and is identical to the result of synchronized and centralized SGD [Theorem 3.8, (Hardt, Recht, and Singer 2016)]. The latter two terms arise from decentralized communication and asynchronous updates. If we consider the case of synchronization, the last term disappears and the result will be reduced to the stability of D-SGD and is slightly better than the result in [Theorem 1, (Sun, Li, and Wang 2021)]. In practice, however, the constant λ and delay $\overline{\tau}$ are always positive, so these two extra terms do not vanish, which means that although AD-SGD can alleviate the communication and synchronization overhead of a distributed training system, it hurts algorithmic stability.

Notice that the learning rate α_t is a vital parameter in Theorem 1, where a small learning rate can mitigate the negative effects of decentralization and asynchronous delays on stability. We further investigate the stability results for two commonly used learning rates in the following corollary.

Corollary 1 *Let the loss function be convex, and Assumptions 1-3 hold. For the constant learning rate $\alpha_t = \alpha$, we have*

$$\epsilon_{\text{stab}} \le \frac{2L^2\alpha T}{nm} + \frac{2\beta L^2\alpha^2 T}{m}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big).$$

If we choose the decreasing learning rate $\alpha_t = \frac{1}{t+1}$,

$$\epsilon_{\text{stab}} \le \frac{2L^2}{nm}\ln T + \frac{2\beta L^2}{m}\Big(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\Big),$$

where $C_\lambda = \frac{8}{\lambda e^2 \ln^2 \frac{1}{\lambda}} + \frac{2}{\lambda \ln \frac{1}{\lambda}}$ *is a constant determined by* λ*, and* $\overline{\tau}$ *is the delay bound.*

Corollary 1 shows that AD-SGD also has the *train faster, generalize better* property as (Hardt, Recht, and Singer 2016) in the convex case. Therefore, in practice, stopping training early after reaching a low training error can be considered to achieve a better generalization performance.

Strongly Convex Case

In the strongly convex case, the gradient update has a well-contracted property, which gifts the following better stability bound.

Theorem 2 *Let the loss function be* μ*-strongly convex, and Assumptions 1-3 hold. If we run AD-SGD with the constant learning rate* $\alpha_t \equiv \alpha \leq m/\beta$ *for* T *iterations, then the uniform stability satisfies*

$$\epsilon_{\text{stab}} \leq \frac{2L^2}{\mu n} + \frac{2\beta L^2 \alpha}{\mu}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big).$$

Furthermore, if we choose the decreasing learning rate $\alpha_t = \frac{m}{\mu(t+1)}$*, AD-SGD has the uniform stability with*

$$\epsilon_{\text{stab}} \leq \frac{2L^2}{\mu n} + \frac{2\beta L^2 (mC_\lambda + \overline{\tau}^2 \lambda^{\overline{\tau}})}{\mu^2 \lambda^{\overline{\tau}}} \frac{\ln T + 1}{T}.$$

Compared with the convex case, the uniform stability ϵ_{stab} is independent of the iterative number T with the constant learning rate. Furthermore, the extra errors introduced by asynchrony and decentralization in the decreasing learning rate strategy vanish as training proceeds.

Non-convex Case

This part focus on the non-convex optimization problem, which is widespread in training deep neural networks.

Theorem 3 *Assume that the loss function is* L*-Lipschitz and* β*-smooth, then the uniform stability* ϵ_{stab} *of AD-SGD after* T *iterations is bounded by*

$$\sum_{t=1}^{T-1} \Big[\prod_{k=t+1}^{T-1} (1+\frac{\beta\alpha_k}{m}) \Big] \frac{2L^2 \alpha_t}{m} \Big(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1} \beta \alpha_s \lambda^{t-\tau_t-1-s} \Big).$$

Without the convexity property, this bound expands with coefficient $(1 + \beta\alpha/m)$, indicating that the algorithmic stability deteriorates when optimizing non-convex problems. The last term reflects the joint effect of decentralization, asynchrony, and learning rate on the stability of AD-SGD, which can be reduced by setting appropriate learning rate and delays. However, decentralization always compromises the stability since $0 < \lambda < 1$. We then need to construct a well-connected communication graph (means a smaller λ) in practice for better generalizability, which is consistent with the findings in (Zhu et al. 2022). Corollary 2 gives the simplified results about two common learning rates.

Corollary 2 *Let Assumptions 1-3 hold. If we run AD-SGD with the constant learning rate* $\alpha_t \equiv \alpha$ *for* T *iterations, then*

$$\epsilon_{\text{stab}} \leq \frac{2L^2(1+\beta n\alpha - \lambda)}{\beta n(1-\lambda)} (1 + \frac{\beta\alpha}{m})^{T-1}.$$

If the learning rate is decreasing by $\alpha_t = \frac{m}{\beta(t+1)}$*, we have*

$$\epsilon_{\text{stab}} \leq \frac{2L^2(1+nm-\lambda)}{\beta n(1-\lambda)} T.$$

Following [Lemma 3.11, (Hardt, Recht, and Singer 2016)], we can improve the upper bound on the stability of AD-SGD by considering the timing of encountering different samples in datasets $\mathcal{S}$ and $\mathcal{S}'$ during the training.

Theorem 4 *Let the loss function* $f(\cdot; \mathbf{z}) \in [0,1]$ *and Assumptions 1-3 hold. If we set* $\alpha_t = \frac{mc}{t+1}$ *where constant* c *is small enough. Then the uniform stability satisfies*

$$\epsilon_{\text{stab}} \leq \frac{1 + 1/\beta c}{n} \left[2L^2 c (1 + \frac{nm\beta c}{1-\lambda}) \right]^{\frac{1}{\beta c+1}} T^{\frac{\beta c}{\beta c+1}}.$$

Excess Generalization Error of AD-SGD

In this section, we concentrate on the excess generalization error, which captures the performance of the output model on unknown data (i.e., model's generalization performance). From decomposition (1), excess generalization error ϵ_{exc} is upper bounded by the summation of generalization error ϵ_{gen} and optimization error ϵ_{opt}, where ϵ_{gen} is controlled by the uniform stability ϵ_{stab} in the previous section.

Although there have been many optimization analyses of AD-SGD (Lian et al. 2018; Nadiradze et al. 2021; Jiang et al. 2021; Xu, Zhang, and Wang 2021), they only give convergence results for the gradient norm $\mathbb{E}\|\nabla F_{\mathcal{S}}(\mathbf{x})\|$. In this study, we need to investigate the optimization error of the objective function value $\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})]$, which is still absent for AD-SGD. We first give the optimization error in terms of objective function value for the strongly convex, convex, and non-convex cases. Then we derive the excess generalization error ϵ_{exc} of AD-SGD. We only present the results for the decreasing learning rate due to space restrictions. More theoretical analysis is included in the supplementary material.

Strongly Convex Optimization

Theorem 5 *Let the loss function be* μ*-strongly convex, and Assumptions 1-4 hold. If the learning rate is chosen as* $\alpha_t = \frac{m}{2\mu(t+1)}$*, we have*

$$\epsilon_{\text{opt}} \leq \frac{\beta L^2 \ln T}{8\mu^2 T} + \frac{\beta^2 r L (mC_\lambda + \overline{\tau}^2 \lambda^{\overline{\tau}})}{2\mu^2 \lambda^{\overline{\tau}}} \frac{\ln T + 1}{T} + \frac{2\beta r^2}{T}.$$

where r *is the radius of the closed ball in Assumption 4. Then, the excess generalization error satisfies*

$$\begin{aligned}\epsilon_{\text{exc}} \leq & \frac{2L^2}{\mu n} + \frac{\beta L(4L + \beta r)(mC_\lambda + \overline{\tau}^2 \lambda^{\overline{\tau}})}{2\mu^2 \lambda^{\overline{\tau}}} \frac{\ln T + 1}{T} \\ & + \frac{\beta L^2 \ln T}{8\mu^2 T} + \frac{\beta \|\mathbf{x}_1 - \mathbf{x}^*_{\mathcal{S}}\|}{2T}.\end{aligned}$$

Theorem 5 shows that asynchrony and decentralization also impair the optimization process of AD-SGD, but the optimization error vanishes as the training proceeds. Therefore, we can make a trade-off between the optimization and generalization errors for the total training iterations T to minimize the excess generalization error ϵ_{exc}. Also, similar to

(Kuzborskij and Lampert 2018; Lei and Ying 2020), Theorem 5 reveals the effect of the initial point on the generalization performance, i.e., the generalizability improves if we start with a good model.

Convex Optimization

Without the strong convexity, the optimization analysis often turns to the average model $\overline{\mathbf{x}}_T = \frac{\sum_{t=1}^{T} \alpha_t \mathbf{x}_t}{\sum_{t=1}^{T} \alpha_t}$ (Sun, Li, and Wang 2021; Lei and Ying 2020).

Theorem 6 *Let the loss function be convex and Assumptions 1-4 hold. With a decreasing learning rate $\alpha_t = \frac{1}{t+1}$, the optimization error satisfies*

$$\epsilon_{\text{opt}} \leq \Big[4mr^2 + 4\beta r L(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{L^2}{m}\Big] \frac{1}{\ln(T+1)}.$$

For consistency, we first need to give the uniform stability of the average model $\overline{\mathbf{x}}_T$, denoted as $\epsilon_{\text{ave}-\text{stab}}$, and then the excess generalization error bound follows.

Theorem 7 *Let the loss function be convex and Assumptions 1-4 hold. If the learning rate is chosen as $\alpha_t = \frac{1}{t+1}$, we have*

$$\epsilon_{\text{ave}-\text{stab}} \leq \frac{2L^2}{nm} \ln(T+1) + \frac{4\beta L^2}{m} \left(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\right).$$

Then, the excess generalization error is bounded by

$$\epsilon_{\text{exc}} \leq \frac{2L^2}{nm} \ln(T+1) + \frac{4\beta L^2}{m} \left(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\right) + \Big[m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\| + 4\beta r L(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{L^2}{m}\Big] \frac{1}{\ln(T+1)}.$$

Non-convex Optimization

The analysis in non-convex settings is more challenging but also more important since optimization problems in the ML community are usually non-convex. In this part, we provide optimization error results for non-convex problems under the following Polyak-Łojasiewicz (PŁ) condition (Polyak 1963; Lojasiewicz 1963).

Definition 2 (PŁ-condition) *Let $\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x}} f(\mathbf{x})$. We say that a function $f(\mathbf{x}) : \Omega \to \mathbb{R}$ satisfies the γ-PŁ condition, where $\gamma > 0$ is a constant, if for $\forall \mathbf{x} \in \Omega$, we have*

$$2\gamma[f(\mathbf{x}) - f(\mathbf{x}^*)] \leq \|\nabla f(\mathbf{x})\|^2.$$

This condition reveals the relationship between the loss function value and its gradient norm. The PŁ-condition (a.k.a., gradient-dominated condition) is widely adopted in the convergence and generalization analysis for non-convex optimization and was shown to hold true for deep (linear) and shallow neural networks (Charles and Papailiopoulos 2018; Karimi, Nutini, and Schmidt 2016; Lei, Ledent, and Kloft 2020; Zhou, Liang, and Zhang 2022).

Theorem 8 *Suppose that the loss function satisfies the γ-PŁ condition and Assumptions 1-4 hold. If we run AD-SGD with $\alpha_t = \frac{m}{\gamma(t+1)}$ for T iterations, the optimization error satisfies*

$$\epsilon_{\text{opt}} \leq \Big[2Lr + \frac{\beta m L^2}{\gamma^2}(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{\beta L^2}{2\gamma^2}\Big] \frac{1}{\ln(T+1)}.$$

The uniform stability of the average model and the excess generalization error of AD-SGD in the non-convex case are bounded in the following theorem.

Theorem 9 *Let the loss function satisfies the γ-PŁ condition and Assumptions 1-4 hold. If the learning rate is chosen as $\alpha_t = \frac{mc}{t+1}$ with a small constant c, we have*

$$\epsilon_{\text{ave}-\text{stab}} \leq \frac{4L^2}{\beta c} \left(\frac{1}{\beta n} + \frac{mc}{1-\lambda}\right) \frac{(T+1)^{\beta c}}{\ln(T+1)}.$$

Then, the excess generalization error satisfies

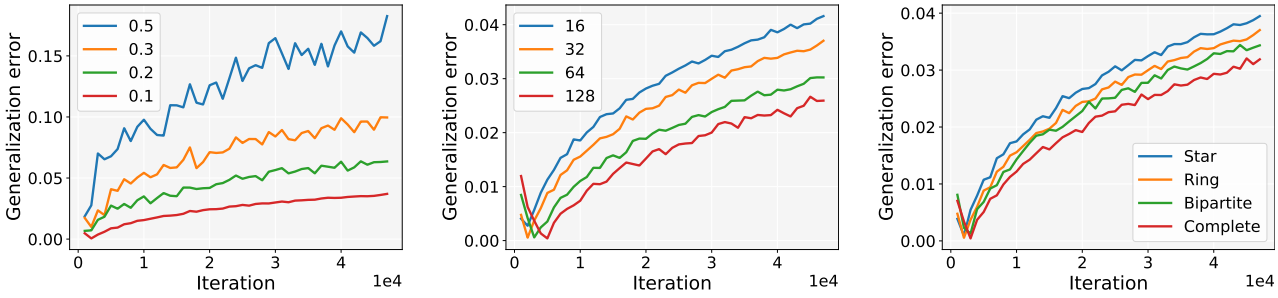
$$\epsilon_{\text{exc}} \leq \frac{4L^2}{\beta c} \left(\frac{1}{\beta n} + \frac{mc}{1-\lambda}\right) \frac{(T+1)^{\beta c}}{\ln(T+1)} + \Big[2Lr + \beta m L^2 c^2 (\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{\beta L^2 c^2}{2}\Big] \frac{1}{\gamma c \ln(T+1)}.$$

Experiment

This section contains extensive experiments to measure the generalization performance of the AD-SGD algorithm. We first use the convex linear model classifying the MNIST (LeCun et al. 1998) dataset to verify the theoretical results in the general convex case. For the non-convex problem, we conducted abundant experiments with the deep models ResNet-18 (He et al. 2016) and VGG-16 (Simonyan and Zisserman 2015) on three commonly used datasets, CIFAR-10, CIFAR-100 (Krizhevsky, Hinton et al. 2009), and Tiny-ImageNet (Le and Yang 2015). The local training batch size is set to 256 for all experiments. We focus on exploring the role played by learning rates, asynchronous delays, and decentralized topologies. To make the results more interpretable, we avoid other training techniques such as warm-up or weight decay.

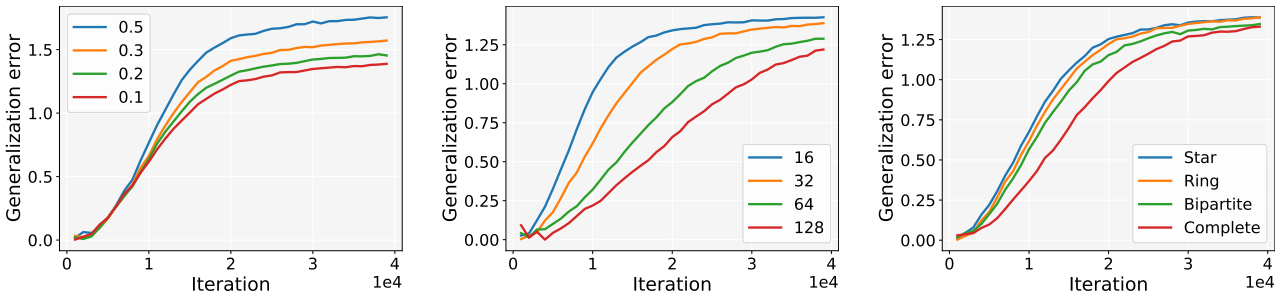
The experiments are conducted on four physical machines with a total of 16 distributed computing workers. Each machine is equipped with four Nvidia RTX-3090 24 GB GPUs, two Intel Xeon 4214R @2.40 GHz CPUs and 128 GB DDR4 RAMs, and the machines are connected via 100 Gbps InfiniBand. All our experimental results are based on a PyTorch (Paszke et al. 2019) implementation of the NCCL backend.

Figures 1 and 2 illustrate the experimental results of classifying MNIST with a convex linear model and CIFAR-100 with the non-convex ResNet-18. The results show that the generalization error increases as training proceeds, which is consistent with our theoretical findings. To explore the effect of varying learning rates on the generalization performance, we fixed the delay sequence and the decentralized topology, and the results are presented in Figures 1(a) and 2(a). According to our theoretical results, increasing the learning rate will impair the stability of AD-SGD and thus leads to a larger generalization error, which is verified by the experimental observations. We then fixed the learning rate and communication topology to perform AD-SGD with different asynchronous delays. Figures 1(b) and 2(b) show that a reasonable increase in asynchronous delay reduces the generalization error, which is consistent with Theorem 3 but also implies the theoretical results regarding asynchrony are pessimistic in the convex case.



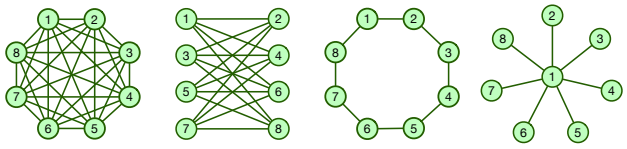
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 1: Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies when optimizing general *convex* models. Here generalization error is the absolute value of the difference between testing and training errors. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 2: Generalization error for optimizing *non-convex* problems. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha_t = \frac{0.1}{1+0.01t}$, ring topology; (c). Fixed $\alpha_t = \frac{0.1}{1+0.01t}, \overline{\tau} = 32$.



(a) Complete (b) Bipartite (c) Ring (d) Star

Figure 3: Illustration of the communication topologies.

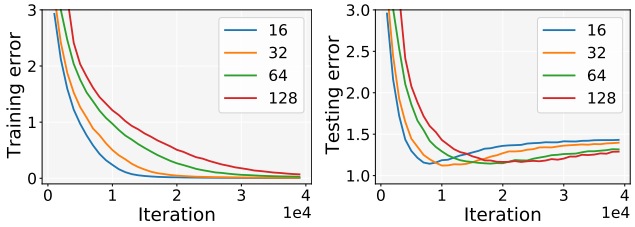


Figure 4: Training and testing errors for varying delays.

The four communication topologies used in the experiments are shown in Figure 3, where the *complete* topology is taken as the baseline because it is not a decentralized setup. The values of λ characterize the sparsity of the four topologies and satisfy the following relationship

$$0 = \lambda_{\text{complete}} < \lambda_{\text{bipartite}} < \lambda_{\text{ring}} \approx \lambda_{\text{star}} < 1.$$

Theoretical analysis shows that a well-connected communication topology (implying a smaller λ) can improve its generalization performance. Therefore, although decentralization reduces the communication overhead of the AD-SGD algorithm, it impairs the generalization performance, which is also demonstrated in the results of Figures 1(c) and 2(c).

Figure 4 presents the training and testing errors of AD-SGD, corresponding to its optimization error and excess generalization error. The experimental configuration is the same as in Figure 2(b). Figure 4(a) shows that the asynchronous delay impairs the training process, which is consistent with our optimization theory. On the other hand, appropriately increasing the delay can alleviate the overfitting phenomenon, as shown in Figure 4(b), which also explains why adding asynchronous delay can reduce the generalization error. More experimental results, including the performance of convex models under decreasing learning rate strategies; non-convex ResNet-18 and VGG-16 on the CIFAR-10, CIFAR-100, and Tiny-ImageNet datasets, are included in the supplementary material.

Conclusion

In this paper, we study the generalization performance of the asynchronous decentralized stochastic gradient descent algorithm for the first time. Building on the uniform stability framework, we present upper bounds for the generalization error and excess generalization error of AD-SGD in the strongly convex, convex and non-convex cases. Our results reveal the effects of asynchronous delay, decentralized topology, learning rate and training iterations on the generalizability of AD-SGD, and extensive experiments are conducted to verify the theoretical findings. Future research directions include deriving sharper upper bounds of stability and generalizability, and lower bounds on the generalization error of SGD in the asynchronous decentralized setting.

Acknowledgments

The authors thank all the anonymous reviewers for their valuable feedback. This work is partly sponsored by the National Science Foundation of China (62025208) and the Hunan Provincial Natural Science Foundation of China (2022JJ10065).

References

Agarwal, A.; and Duchi, J. C. 2011. Distributed delayed stochastic optimization. In *Advances in Neural Information Processing Systems*, volume 24.

Assran, M.; Aytekin, A.; Feyzmahdavian, H. R.; Johansson, M.; and Rabbat, M. G. 2020. Advances in asynchronous parallel and distributed optimization. *Proceedings of the IEEE*, 108(11): 2013–2031.

Assran, M.; Loizou, N.; Ballas, N.; and Rabbat, M. 2019. Stochastic gradient push for distributed deep learning. In *International Conference on Machine Learning*, 344–353. PMLR.

Bassily, R.; Feldman, V.; Guzmán, C.; and Talwar, K. 2020. Stability of stochastic gradient descent on nonsmooth convex losses. In *Advances in Neural Information Processing Systems*, volume 33, 4381–4391.

Bousquet, O.; and Elisseeff, A. 2002. Stability and generalization. *Journal of Machine Learning Research*, 2: 499–526.

Charles, Z.; and Papailiopoulos, D. 2018. Stability and generalization of learning algorithms that converge to global optima. In *International Conference on Machine Learning*, volume 80, 745–754. PMLR.

Cui, X.; Zhang, W.; Kayi, A.; Liu, M.; Finkler, U.; Kingsbury, B.; Saon, G.; and Kung, D. 2021. Asynchronous decentralized distributed training of acoustic models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29: 3565–3576.

Dean, J.; Corrado, G.; Monga, R.; Chen, K.; Devin, M.; Mao, M.; Ranzato, M.; Senior, A.; Tucker, P.; Yang, K.; et al. 2012. Large scale distributed deep networks. In *Advances in Neural Information Processing Systems*, 1223–1231.

Elisseeff, A.; Evgeniou, T.; Pontil, M.; and Kaelbing, L. P. 2005. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(1).

Hardt, M.; Recht, B.; and Singer, Y. 2016. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, 1225–1234. PMLR.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Jiang, J.; Zhang, W.; Gu, J.; and Zhu, W. 2021. Asynchronous decentralized online learning. In *Advances in Neural Information Processing Systems*, volume 34, 20185–20196.

Karimi, H.; Nutini, J.; and Schmidt, M. 2016. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, 795–811. Springer.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images. 32–33.

Kuzborskij, I.; and Lampert, C. 2018. Data-dependent stability of stochastic gradient descent. In *International Conference on Machine Learning*, 2815–2824. PMLR.

Lan, G.; and Zhou, Y. 2021. Asynchronous decentralized accelerated stochastic gradient descent. *IEEE Journal on Selected Areas in Information Theory*, 2(2): 802–811.

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *CS 231N*.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Lei, Y.; Ledent, A.; and Kloft, M. 2020. Sharper generalization bounds for pairwise learning. In *Advances in Neural Information Processing Systems*, volume 33, 21236–21246.

Lei, Y.; Yang, Z.; Yang, T.; and Ying, Y. 2021. Stability and generalization of stochastic gradient methods for minimax problems. In *International Conference on Machine Learning*, 6175–6186. PMLR.

Lei, Y.; and Ying, Y. 2020. Fine-grained analysis of stability and generalization for stochastic gradient descent. In *International Conference on Machine Learning*, 5809–5819. PMLR.

Li, M.; Andersen, D. G.; Park, J. W.; Smola, A. J.; Ahmed, A.; Josifovski, V.; Long, J.; Shekita, E. J.; and Su, B.-Y. 2014. Scaling distributed machine learning with the parameter server. In *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*, 583–598.

Lian, X.; Huang, Y.; Li, Y.; and Liu, J. 2015. Asynchronous parallel stochastic gradient for nonconvex optimization. In *Advances in Neural Information Processing Systems*, volume 28.

Lian, X.; Zhang, C.; Zhang, H.; Hsieh, C.-J.; Zhang, W.; and Liu, J. 2017. Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 30.

Lian, X.; Zhang, W.; Zhang, C.; and Liu, J. 2018. Asynchronous decentralized parallel stochastic gradient descent.

In *International Conference on Machine Learning*, 3043–3052. PMLR.

Lojasiewicz, S. 1963. A topological property of real analytic subsets. *Coll. du CNRS, Les équations aux dérivées partielles*, 117(87-89): 2.

Luo, Q.; He, J.; Zhuo, Y.; and Qian, X. 2020. Prague: High-performance heterogeneity-aware asynchronous decentralized training. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, 401–416.

Mou, W.; Wang, L.; Zhai, X.; and Zheng, K. 2018. Generalization bounds of sgld for non-convex learning: Two theoretical viewpoints. In *Conference on Learning Theory*, 605–638. PMLR.

Nadiradze, G.; Sabour, A.; Davies, P.; Li, S.; and Alistarh, D. 2021. Asynchronous decentralized SGD with quantized and local updates. In *Advances in Neural Information Processing Systems*, volume 34, 6829–6842.

Nedic, A.; and Ozdaglar, A. 2009. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1): 48–61.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 8024–8035.

Polyak, B. T. 1963. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4): 864–878.

Recht, B.; Re, C.; Wright, S.; and Niu, F. 2011. Hogwild!: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 24.

Regatti, J.; Tendolkar, G.; Zhou, Y.; Gupta, A.; and Liang, Y. 2019. Distributed SGD generalizes well under asynchrony. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 863–870. IEEE.

Robbins, H.; and Monro, S. 1951. A stochastic approximation method. *The Annals of Mathematical Statistics*, 400–407.

Shalev-Shwartz, S.; Shamir, O.; Srebro, N.; and Sridharan, K. 2010. Learnability, stability and uniform convergence. *Journal of Machine Learning Research*, 11: 2635–2670.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Sirb, B.; and Ye, X. 2016. Consensus optimization with delayed and stochastic gradients on decentralized networks. In *IEEE International Conference on Big Data*, 76–85. IEEE.

Sun, T.; and Li, D. 2020. Capri: Consensus accelerated proximal reweighted iteration for a class of nonconvex minimizations. *IEEE Transactions on Knowledge and Data Engineering*.

Sun, T.; Li, D.; and Wang, B. 2021. Stability and generalization of decentralized stochastic gradient descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 9756–9764.

Sun, T.; Li, D.; and Wang, B. 2022. Decentralized federated averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Tang, H.; Lian, X.; Yan, M.; Zhang, C.; and Liu, J. 2018. D^2: Decentralized training over decentralized data. In *International Conference on Machine Learning*, volume 80, 4848–4856. PMLR.

Tsitsiklis, J.; Bertsekas, D.; and Athans, M. 1986. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9): 803–812.

Tsitsiklis, J. N. 1984. Problems in decentralized decision making and computation. Technical report, Massachusetts Inst of Tech Cambridge Lab for Information and Decision Systems.

Wu, X.; Zhang, J.; and Wang, F.-Y. 2019. Stability-based generalization analysis of distributed learning algorithms for big data. *IEEE Transactions on Neural Networks and Learning Systems*, 31(3): 801–812.

Xing, Y.; Song, Q.; and Cheng, G. 2021. On the algorithmic stability of adversarial training. In *Advances in Neural Information Processing Systems*, volume 34, 26523–26535.

Xu, J.; Zhang, W.; and Wang, F. 2021. A(DP)^{2}SGD: Asynchronous decentralized parallel stochastic gradient descent with differential privacy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2021. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3): 107–115.

Zhang, Y.; Zhang, W.; Bald, S.; Pingali, V. P.; Chen, C.; and Goswami, M. 2022. Stability of SGD: Tightness analysis and improved bounds. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

Zheng, S.; Meng, Q.; Wang, T.; Chen, W.; Yu, N.; Ma, Z.-M.; and Liu, T.-Y. 2017. Asynchronous stochastic gradient descent with delay compensation. In *International Conference on Machine Learning*, 4120–4129. PMLR.

Zhou, Y.; Liang, Y.; and Zhang, H. 2022. Understanding generalization error of SGD in nonconvex optimization. *Machine Learning*, 111(1): 345–375.

Zhu, T.; He, F.; Zhang, L.; Niu, Z.; Song, M.; and Tao, D. 2022. Topology-aware generalization of decentralized SGD. In *International Conference on Machine Learning*. PMLR.

Zinkevich, M.; Weimer, M.; Li, L.; and Smola, A. 2010. Parallelized stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 23.

Supplementary Materials for

Stability-Based Generalization Analysis of the Asynchronous Decentralized SGD

The supplementary material contains the full experimental results and detailed proofs of our theoretical findings.

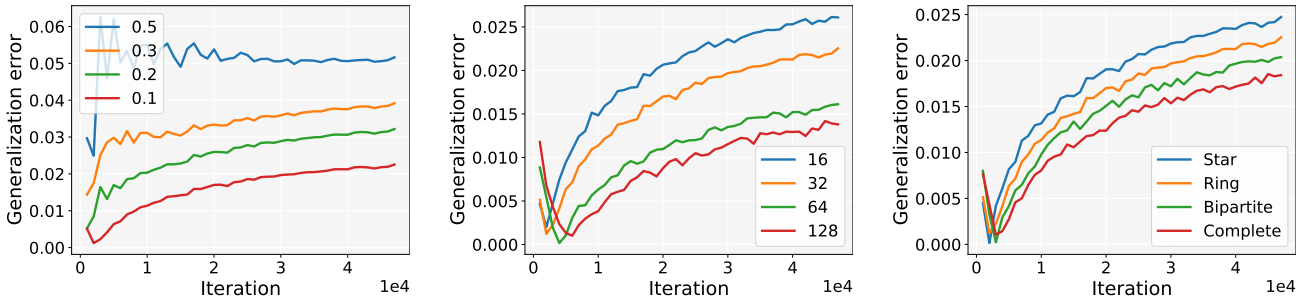
Contents

A More Experimental Results

In AD-SGD (Lian et al. 2018), the communication topology is designed as a bipartite graph in order to prevent the deadlock problem. The topologies that we have employed (as shown in Figure 3) all satisfy this property. Consider a distributed system with 16 computing workers, the corresponding doubly stochastic matrix of the four topologies are

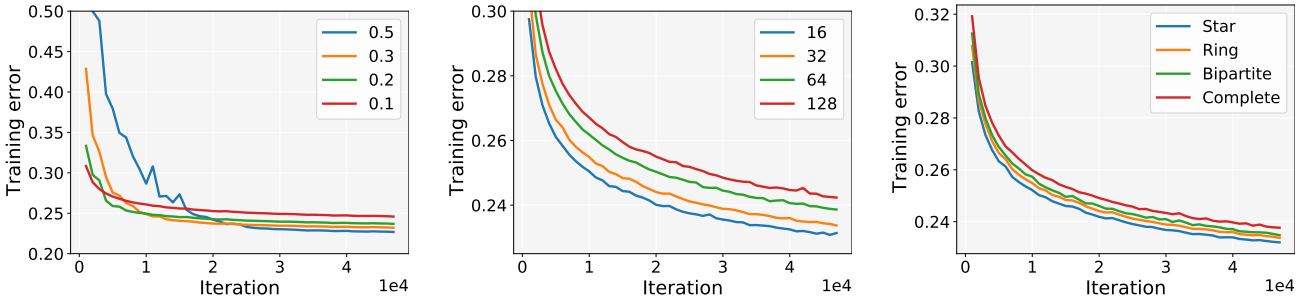
$$\mathbf{W}_{\text{comp}} = \begin{pmatrix} \frac{1}{16} & \cdots & \frac{1}{16} \\ \frac{1}{16} & \cdots & \frac{1}{16} \\ \vdots & \ddots & \vdots \\ \frac{1}{16} & \cdots & \frac{1}{16} \\ \frac{1}{16} & \cdots & \frac{1}{16} \end{pmatrix} \mathbf{W}_{\text{bipa}} = \begin{pmatrix} \frac{1}{9} & \frac{1}{9} & \cdots & 0 & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \cdots & \frac{1}{9} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \frac{1}{9} & \cdots & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & 0 & \cdots & \frac{1}{9} & \frac{1}{9} \end{pmatrix} \mathbf{W}_{\text{ring}} = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \cdots & 0 & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & 0 & \cdots & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \mathbf{W}_{\text{star}} = \begin{pmatrix} \frac{1}{16} & \frac{1}{16} & \cdots & \frac{1}{16} \\ \frac{1}{16} & \frac{15}{16} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{16} & 0 & \cdots & 0 \\ \frac{1}{16} & 0 & \cdots & \frac{15}{16} \end{pmatrix}$$

In the following, we will show more experimental results, including the performance of convex models with decreasing learning rate; non-convex ResNet-18 and VGG-16 on the CIFAR-10, CIFAR-100, and Tiny-ImageNet datasets. The experimental observations are consistent with the theoretical analysis and description of the experimental results in the main text.



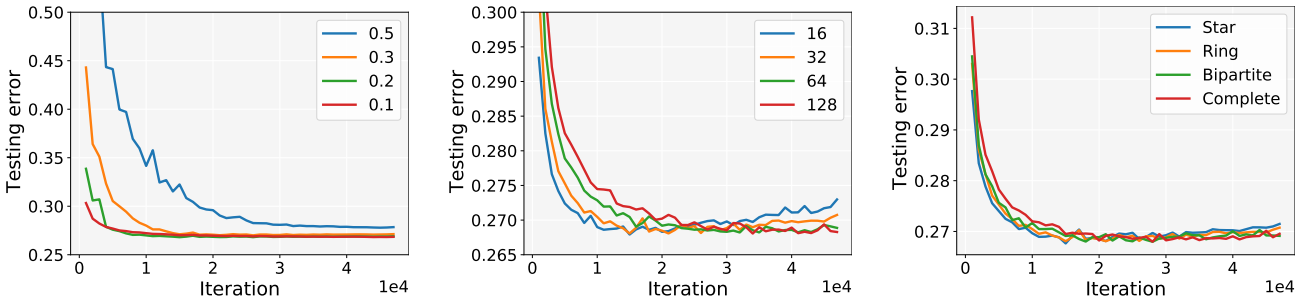
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 1: Convex model on the MNIST dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies with the decreasing learning rate. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha_t = \frac{0.1}{1+0.01t}$, ring topology; (c). Fixed $\alpha_t = \frac{0.1}{1+0.01t}, \overline{\tau} = 32$.



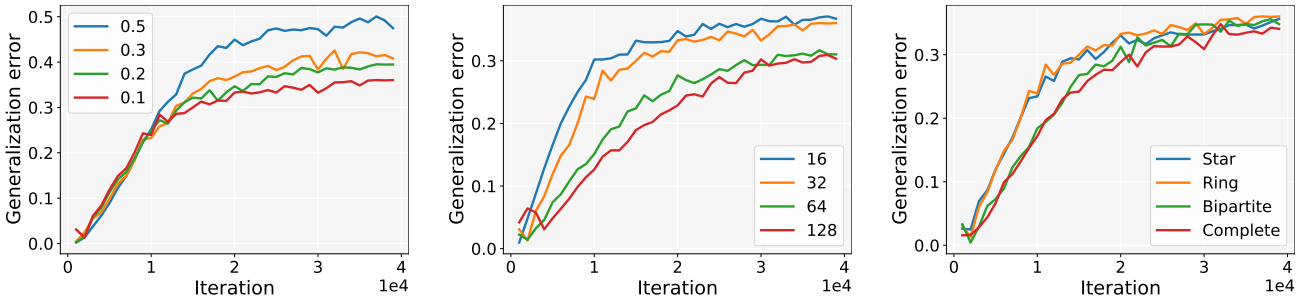
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 2: Convex model on the MNIST dataset. Training errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



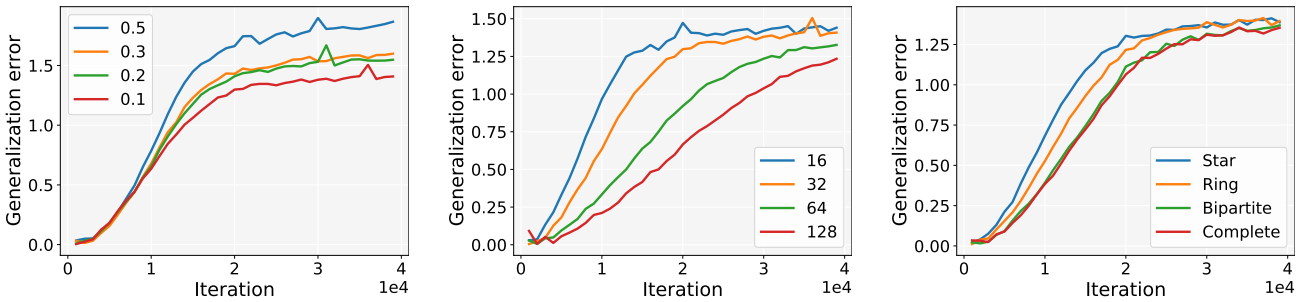
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 3: Convex model on the MNIST dataset. Testing errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



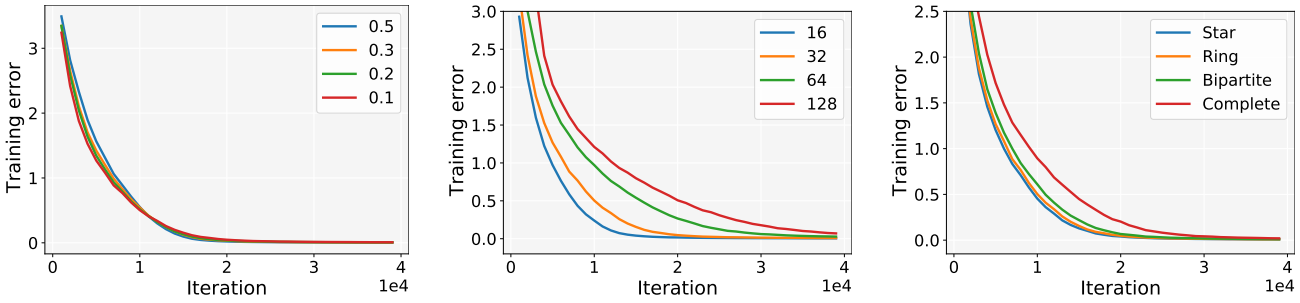
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 4: Non-convex ResNet-18 on the CIFAR-10 dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



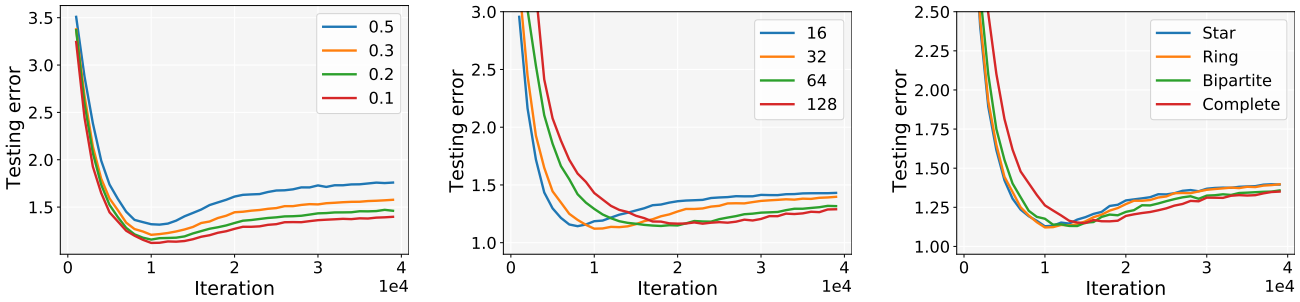
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 5: Non-convex ResNet-18 on the CIFAR-100 dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



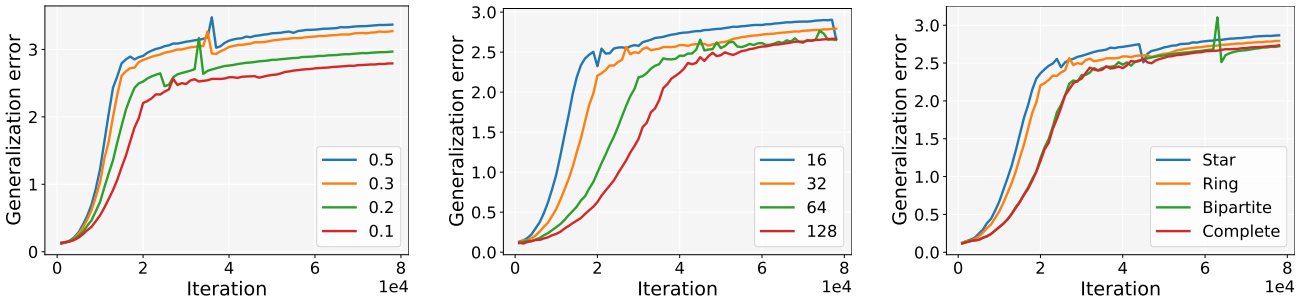
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 6: Non-convex ResNet-18 on the CIFAR-100 dataset. Training errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha_t = \frac{0.1}{1+0.01t}$, ring topology; (c). Fixed $\alpha_t = \frac{0.1}{1+0.01t}, \overline{\tau} = 32$.



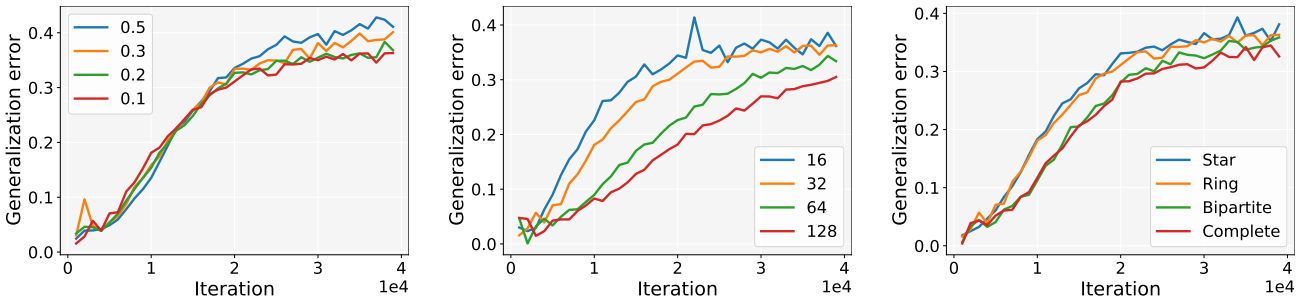
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 7: Non-convex ResNet-18 on the CIFAR-100 dataset. Testing errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology. Decreasing learning rate $\alpha_t = \frac{\alpha}{1+0.01t}$ with varying α; (b). Fixed $\alpha_t = \frac{0.1}{1+0.01t}$, ring topology; (c). Fixed $\alpha_t = \frac{0.1}{1+0.01t}, \overline{\tau} = 32$.



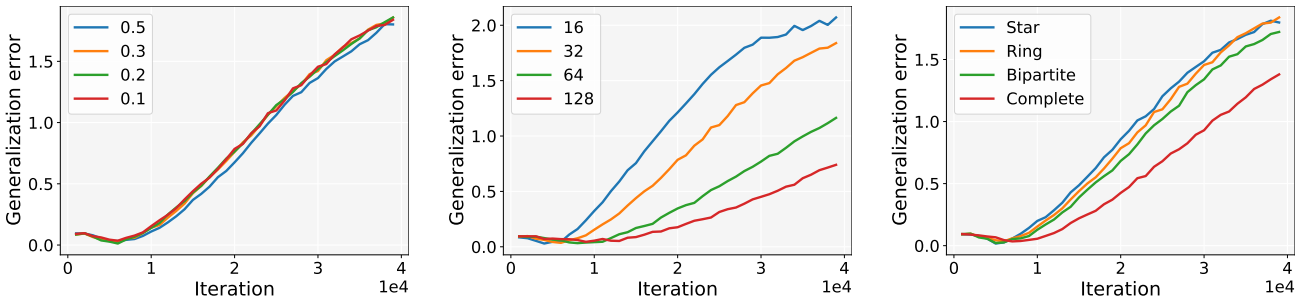
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 8: Non-convex ResNet-18 on the Tiny-ImageNet dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



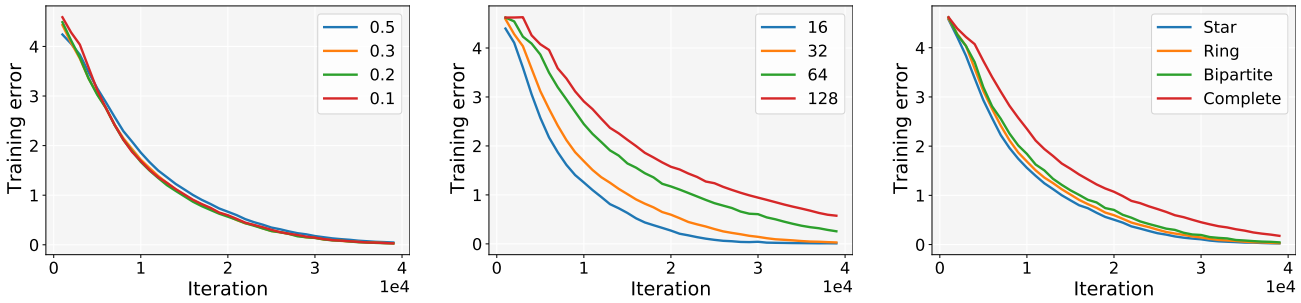
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 9: Non-convex VGG-16 on the CIFAR-10 dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



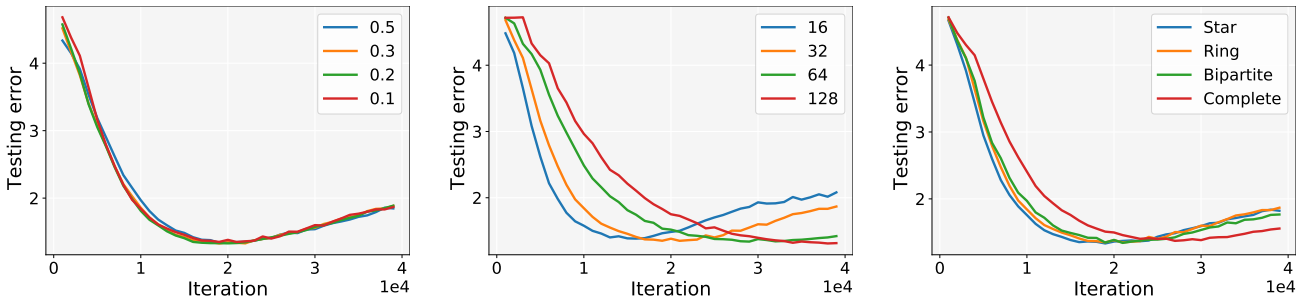
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 10: Non-convex VGG-16 on the CIFAR-100 dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



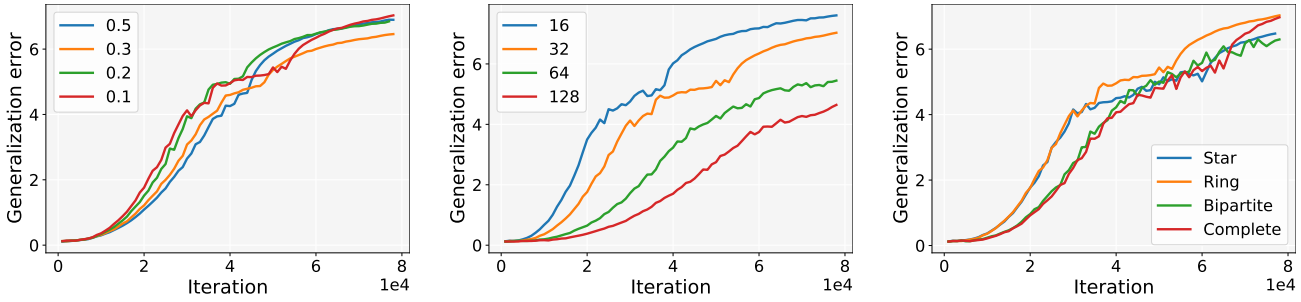
(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 11: Non-convex VGG-16 on the CIFAR-100 dataset. Training errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 12: Non-convex VGG-16 on the CIFAR-100 dataset. Testing errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.



(a) Learning rate (b) Asynchronous delay (c) Decentralized topology

Figure 13: Non-convex VGG-16 on the Tiny-ImageNet dataset. Generalization errors for varying learning rates, asynchronous delays, and decentralized topologies. (a). Fixed maximum delay $\overline{\tau} = 32$, ring topology; (b). Fixed learning rate $\alpha = 0.1$, ring topology; (c). Fixed $\alpha = 0.1, \overline{\tau} = 32$.

B Missing Theoretical Proofs

B.1 Properties and Technical Lemmas

From the iterative format of AD-SGD, i.e.,

$$\mathbf{X}_{t+1} = \mathbf{X}_t\mathbf{W} - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t; \mathbf{z}_{j_t}), \tag{B.1}$$

the consensus model has the following recursive property

$$\begin{aligned}\mathbf{x}_{t+1} &= \frac{1}{m}\sum_{i=1}^{m}\mathbf{x}_{t+1}(i) = \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{m} w_{i,k}\mathbf{x}_t(k) - \alpha_t\frac{1}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)}) \\ &= \frac{1}{m}\sum_{i=1}^{m}\mathbf{x}_t(i) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)}) \\ &= \mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)}).\end{aligned} \tag{B.2}$$

Lemma 4 (Lemma 3.7, (Hardt, Recht, and Singer 2016)) *The following properties hold for every* $\mathbf{z}$.

1. *Assume that* f *is* β*-smooth. Then*

$$\left\|\mathbf{x} - \frac{\alpha}{m}\nabla f(\mathbf{x}; \mathbf{z}) - \mathbf{x}' + \frac{\alpha}{m}\nabla f(\mathbf{x}'; \mathbf{z})\right\| \leq (1 + \frac{\beta\alpha}{m})\|\mathbf{x} - \mathbf{x}'\|. \tag{B.3}$$

2. *Assume that* f *is* β*-smooth, convex. Then for any* $\alpha \leq 2m/\beta$

$$\left\|\mathbf{x} - \frac{\alpha}{m}\nabla f(\mathbf{x}; \mathbf{z}) - \mathbf{x}' + \frac{\alpha}{m}\nabla f(\mathbf{x}'; \mathbf{z})\right\| \leq \|\mathbf{x} - \mathbf{x}'\|. \tag{B.4}$$

3. *Assume that* f *is* β*-smooth,* μ*-strongly convex. Then for any* $\alpha \leq m/\beta$

$$\left\|\mathbf{x} - \frac{\alpha}{m}\nabla f(\mathbf{x}; \mathbf{z}) - \mathbf{x}' + \frac{\alpha}{m}\nabla f(\mathbf{x}'; \mathbf{z})\right\| \leq (1 - \frac{\mu\alpha}{m})\|\mathbf{x} - \mathbf{x}'\|. \tag{B.5}$$

Lemma 5 *For any* $0 < \lambda < 1$ *and* $t \in \mathbb{Z}^+$*, it holds*

$$\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} \leq \frac{C_\lambda}{t}, \tag{B.6}$$

where $C_\lambda = \frac{8}{\lambda e^2 \ln^2\frac{1}{\lambda}} + \frac{2}{\lambda \ln\frac{1}{\lambda}}$ *is a constant.*

Proof. The proof is very similar to [Lemma 5, (Sun, Li, and Wang 2021)], and we include a proof for completeness. For any $0 < \lambda < 1, x \in [s, s+1]$, we have that $\frac{\lambda^{t-1-s}}{s+1} \leq \frac{\lambda^{t-1-x}}{x}$. Then

$$\begin{aligned}\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} &\leq \sum_{s=1}^{t-1}\int_s^{s+1}\frac{\lambda^{t-1-x}}{x}dx \leq \lambda^{t-1}\int_1^t\frac{\lambda^{-x}}{x}dx \leq \lambda^{t-1}\int_1^{\frac{t}{2}}\frac{\lambda^{-x}}{x}dx + \lambda^{t-1}\int_{\frac{t}{2}}^t\frac{\lambda^{-x}}{x}dx \\ &\leq \lambda^{\frac{t}{2}-1}\int_1^{\frac{t}{2}}\frac{1}{x}dx + \frac{2\lambda^{t-1}}{t}\int_{\frac{t}{2}}^t\lambda^{-x}dx \leq \lambda^{\frac{t}{2}-1}\ln(\frac{t}{2}) + \frac{2}{t\lambda\ln\frac{1}{\lambda}} \\ &\leq \frac{t\lambda^{\frac{t}{2}-1}}{2} + \frac{2}{t\lambda\ln\frac{1}{\lambda}}.\end{aligned}$$

Now, we provide the bound for $\sup_{t\geq 1}\{t^2\lambda^{\frac{t}{2}-1}\}$. It is easy to check that $t = 4/\ln\frac{1}{\lambda}$ achieves the maximum, which indicates

$$\sup_{t\geq 1}\{t^2\lambda^{\frac{t}{2}-1}\} \leq \frac{16}{\lambda e^2\ln^2\frac{1}{\lambda}}.$$

In conclude, for $0 < \lambda < 1$

$$\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} \leq \left[\frac{8}{\lambda e^2\ln^2\frac{1}{\lambda}} + \frac{2}{\lambda\ln\frac{1}{\lambda}}\right]\frac{1}{t}.$$

We then competed the proof.

■

B.2 Proof of Lemma 2

From the iterative format (B.1) of AD-SGD and the following notation

$$\begin{aligned}\mathbf{X}_t &= [\mathbf{x}_t(1) \quad \mathbf{x}_t(2) \quad \cdots \quad \mathbf{x}_t(m)];\\ \mathbf{G}(\hat{\mathbf{X}}_t;\mathbf{z}_{j_t}) &= [\mathbf{0} \quad \cdots \quad \mathbf{0} \quad \cdots \quad \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) \quad \mathbf{0} \quad \cdots \quad \mathbf{0}],\end{aligned}$$

we have that $\mathbf{x}_t = \frac{\mathbf{X}_t\mathbf{1}_m}{m}$, $\mathbf{x}_t(i) = \mathbf{X}_t\mathbf{e}_i$, where $\mathbf{e}_i$ is the column vector in $\mathbb{R}^m$ whose i-th element is 1. Then we can derive

$$\begin{aligned}\|\mathbf{x}_{t+1} - \mathbf{x}_{t+1}(i)\| &= \left\|\frac{\mathbf{X}_{t+1}\mathbf{1}_m}{m} - \mathbf{X}_{t+1}\mathbf{e}_i\right\|\\ &= \left\|\frac{\mathbf{X}_t\mathbf{W}\mathbf{1}_m - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t;\mathbf{z}_{j_t})\mathbf{1}_m}{m} - (\mathbf{X}_t\mathbf{W}\mathbf{e}_i - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t;\mathbf{z}_{j_t})\mathbf{e}_i)\right\|\\ &= \left\|\frac{\mathbf{X}_t\mathbf{1}_m - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t;\mathbf{z}_{j_t})\mathbf{1}_m}{m} - (\mathbf{X}_t\mathbf{W}\mathbf{e}_i - \alpha_t\mathbf{G}(\hat{\mathbf{X}}_t;\mathbf{z}_{j_t})\mathbf{e}_i)\right\|\\ &= \left\|\frac{\mathbf{X}_1\mathbf{1}_m - \sum_{s=1}^t \alpha_s\mathbf{G}(\hat{\mathbf{X}}_s;\mathbf{z}_{j_s})\mathbf{1}_m}{m} - \left(\mathbf{X}_1\mathbf{W}^t\mathbf{e}_i - \sum_{s=1}^t \alpha_s\mathbf{G}(\hat{\mathbf{X}}_s;\mathbf{z}_{j_s})\mathbf{W}^{t-s}\mathbf{e}_i\right)\right\|\\ &\overset{(a)}{=} \left\|\sum_{s=1}^t \alpha_s\mathbf{G}(\hat{\mathbf{X}}_s;\mathbf{z}_{j_s})\left(\frac{\mathbf{1}_m}{m} - \mathbf{W}^{t-s}\mathbf{e}_i\right)\right\|\\ &\overset{(b)}{\leq} L\sum_{s=1}^t \alpha_s\left\|\frac{\mathbf{1}_m}{m} - \mathbf{W}^{t-s}\mathbf{e}_i\right\|\\ &\overset{(c)}{\leq} L\sum_{s=1}^t \alpha_s\lambda^{t-s},\end{aligned}$$

where (a) uses $\mathbf{x}_1(1) = \mathbf{x}_1(2) = \cdots = \mathbf{x}_1(m)$, which indicates $\mathbf{X}_1\mathbf{W} = \mathbf{X}_1$ $\frac{\mathbf{X}_1\mathbf{1}_m}{m} - \mathbf{X}_1\mathbf{e}_i = 0, \forall\, i$. (b) uses the bounded gradient assumption, and (c) uses the properties of the doubly random matrix $\mathbf{W}$ ([Lemma 3, (Lian et al. 2018)]). Thus

$$\|\mathbf{x}_{t+1} - \mathbf{x}_{t+1}(i)\| \leq L\sum_{s=1}^t \alpha_s\lambda^{t-s}. \tag{B.7}$$

Remark 1 *If $t = 1$, we have that $\|\mathbf{x}_1 - \mathbf{x}_1(i)\| = 0$, then we define $\sum_{s=1}^{t-1}\alpha_s\lambda^{t-s}|_{t=1} = 0$.*

■

B.3 Proof of Lemma 3

$$\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| \leq \sum_{s=t-\tau_t}^{t-1}\|\mathbf{x}_{s+1} - \mathbf{x}_s\| \leq \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\|\nabla f(\mathbf{x}_{s-\tau_s}(i_s);\mathbf{z}_{j_t(i_s)})\| \leq \frac{L}{m}\sum_{s=t-\tau_t}^{t-1}\alpha_s. \tag{B.8}$$

Remark 2 *If $\tau_t = 0$, we have that $\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| = 0$, then we define $\sum_{s=t-\tau_t}^{t-1}\alpha_s|_{\tau_t=0} = 0$.*

■

B.4 Proof of Theorem 1 (generalization error in the convex case)

Let $\mathcal{S} = \{\mathbf{z}_1, \cdots, \mathbf{z}_{j_*}, \cdots, \mathbf{z}_n\}$ and $\mathcal{S}' = \{\mathbf{z}_1, \cdots, \mathbf{z}'_{j_*}, \cdots, \mathbf{z}_n\}$ be two training dataset of size n differing in only a single example $\mathbf{z}_{j_*}$. $\mathbf{x}_T$ and $\mathbf{x}'_T$ denote the output model of running AD-SGD on $\mathcal{S}$ and $\mathcal{S}'$ for T iterations, respectively. For the two data dividing methods, the probability of AD-SGD selecting the same sample in both $\mathcal{S}$ and $\mathcal{S}'$ at the t-th iteration is $1 - \frac{1}{n}$,

i.e., $j_t(i_t) \neq j_*$. Then we have

$$\begin{aligned}
&\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| = \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\leq \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_t(i_t)})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\overset{(a)}{\leq} \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}_{t-\tau_t}(i_t)\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_t - \mathbf{x}'_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}(i_t)\| \\
&\overset{(b)}{\leq} \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2\beta\alpha_t}{m}\frac{L}{m}\sum_{s=t-\tau_t}^{t-1}\alpha_s + \frac{2\beta\alpha_t}{m}L\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} \\
&\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big),
\end{aligned} \tag{B.9}$$

where (a) uses the convexity (B.4) and the β-smoothness assumption; (b) uses inequalities (B.7), (B.8). With probability $\frac{1}{n}$ the selected example is different, i.e., $j_t(i_t) = j_*$. With the bounded gradient assumption, we have

$$\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| = \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_*}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}'_{j_*})\| \leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2L\alpha_t}{m}. \tag{B.10}$$

Denote $\delta_t = \|\mathbf{x}_t - \mathbf{x}'_t\|$, then $\delta_1 = \|\mathbf{x}_1 - \mathbf{x}'_1\| = 0$. With inequalities (B.9) and (B.10), taking expectation of δ_{t+1} with respect to the randomness of the algorithm, we have

$$\begin{aligned}
\mathbb{E}[\delta_{t+1}] &\leq (1 - \frac{1}{n})\mathbb{E}[\delta_t] + (1 - \frac{1}{n})\frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big) + \frac{1}{n}\mathbb{E}[\delta_t] + \frac{1}{n}\frac{2L\alpha_t}{m} \\
&\leq \mathbb{E}[\delta_t] + \frac{2L\alpha_t}{nm} + \frac{2(n-1)\beta L\alpha_t}{nm}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big).
\end{aligned} \tag{B.11}$$

We then have

$$\begin{aligned}
\mathbb{E}[\delta_T] &\leq \frac{2L}{nm}\sum_{t=1}^{T-1}\alpha_t + \frac{2(n-1)\beta L}{nm}\sum_{t=1}^{T-1}\alpha_t\Big[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big] \\
&\leq \frac{2L}{n}\sum_{t=1}^{T-1}\frac{\alpha_t}{m} + 2\beta L\sum_{t=1}^{T-1}\frac{\alpha_t}{m}\Big[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big].
\end{aligned} \tag{B.12}$$

For every $\mathbf{z}$, the L-Lipschitz condition indicate that

$$\mathbb{E}|f(\mathbf{x}_T; \mathbf{z}) - f(\mathbf{x}'_T; \mathbf{z})| \leq L\mathbb{E}[\delta_T] \leq \frac{2L^2}{n}\sum_{t=1}^{T-1}\frac{\alpha_t}{m} + 2\beta L^2\sum_{t=1}^{T-1}\frac{\alpha_t}{m}\Big[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big],$$

which means that the uniform stability satisfies

$$\epsilon_{\text{stab}} \leq \sum_{t=1}^{T-1}\Big[\frac{2L^2\alpha_t}{nm} + \frac{2\beta L^2\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big]. \tag{B.13}$$

■

B.5 Proof of Corollary 1 (generalization error for different learning rate in the convex case)

According to (B.13), for the constant learning rate $\alpha_t = \alpha$, we have

$$\begin{aligned}
\epsilon_{\text{stab}} &\leq \frac{2L^2}{nm}\sum_{t=1}^{T-1}\alpha + 2\beta L^2\sum_{t=1}^{T-1}\frac{\alpha}{m}\Big[\sum_{s=1}^{t-\tau_t-1}\alpha\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha}{m}\Big] \\
&\leq \frac{2L^2\alpha(T-1)}{nm} + \frac{2\beta L^2\alpha^2}{m}\sum_{t=1}^{T-1}\Big(\frac{1}{1-\lambda} + \frac{\tau_t}{m}\Big) \\
&\leq \frac{2L^2\alpha(T-1)}{nm} + \frac{2\beta L^2\alpha^2(T-1)}{m}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big).
\end{aligned}$$

For the decreasing learning rate $\alpha_t = \frac{1}{t+1}$, it follows that

$$\begin{aligned}
\epsilon_{\text{stab}} &\leq \frac{2L^2}{nm}\sum_{t=1}^{T-1}\alpha_t + \frac{2\beta L^2}{m}\sum_{t=1}^{T-1}\alpha_t\Big[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big] \\
&\leq \frac{2L^2}{nm}\sum_{t=1}^{T-1}\frac{1}{t+1} + \frac{2\beta L^2}{m}\sum_{t=1}^{T-1}\frac{1}{t+1}\Big[\frac{1}{\lambda^{\overline{\tau}}}\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} + \sum_{s=t-\tau_t}^{t-1}\frac{1}{m(s+1)}\Big] \\
&\overset{(a)}{\leq} \frac{2L^2}{nm}\ln T + \frac{2\beta L^2}{m}\sum_{t=1}^{T-1}\frac{1}{t+1}\Big[\frac{C_\lambda}{t\lambda^{\overline{\tau}}} + \frac{\tau_t}{m(t-\tau_t+1)}\Big] \\
&\leq \frac{2L^2}{nm}\ln T + \frac{2\beta L^2}{m}\Big[\frac{C_\lambda}{\lambda^{\overline{\tau}}}\sum_{t=1}^{T-1}(\frac{1}{t} - \frac{1}{t+1}) + \frac{1}{m}\sum_{t=1}^{T-1}(\frac{1}{t-\tau_t+1} - \frac{1}{t+1})\Big] \\
&\overset{(b)}{\leq} \frac{2L^2}{nm}\ln T + \frac{2\beta L^2}{m}\Big(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{\overline{\tau}+\ln(\overline{\tau}+1)}{m}\Big) \\
&\leq \frac{2L^2}{nm}\ln T + \frac{2\beta L^2}{m}\Big(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\Big),
\end{aligned}$$

where (a) uses the inequality (B.6) and

$$\sum_{t=1}^{T-1}\frac{1}{t+1} \leq \sum_{t=1}^{T-1}\int_t^{t+1}\frac{1}{x}dx \leq \int_1^T\frac{1}{x}dx \leq \ln T, \tag{B.14}$$

and (b) uses

$$\begin{aligned}
\sum_{t=1}^{T-1}(\frac{1}{t-\tau_t+1} - \frac{1}{t+1}) &\leq \sum_{t=1}^{\overline{\tau}}(1 - \frac{1}{t+1}) + \sum_{t=\overline{\tau}+1}^{T-1}(\frac{1}{t-\overline{\tau}+1} - \frac{1}{t+1}) \\
&\leq \overline{\tau} + \sum_{t=1}^{\overline{\tau}}\frac{1}{t+1} - \sum_{t=T-\overline{\tau}}^{T-1}\frac{1}{t+1} \leq \overline{\tau} + \sum_{t=1}^{\overline{\tau}}\int_t^{t+1}\frac{1}{x}dx \leq \overline{\tau} + \ln(\overline{\tau}+1).
\end{aligned} \tag{B.15}$$

■

B.6 Proof of Theorem 2 (generalization error for different learning rate in the strongly convex case)

$\mathbf{x}_T$ and $\mathbf{x}'_T$ denote the output model of running AD-SGD on $\mathcal{S}$ and $\mathcal{S}'$ for T iterations, respectively. With probability $1 - \frac{1}{n}$, the example selected in $\mathcal{S}$ and $\mathcal{S}'$ is the same at the t-th iteration, i.e., $j_t(i_t) \neq j_*$. Then we have

$$\begin{aligned}
&\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| = \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\leq \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_t(i_t)})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_t(i_t)}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| \\
&\overset{(a)}{\leq} (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}_{t-\tau_t}(i_t)\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_t - \mathbf{x}'_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}(i_t)\| \\
&\overset{(b)}{\leq} (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2\beta\alpha_t}{m}\frac{L}{m}\sum_{s=t-\tau_t}^{t-1}\alpha_s + \frac{2\beta\alpha_t}{m}L\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} \\
&\leq (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big),
\end{aligned} \tag{B.16}$$

where (a) uses the strong convexity (B.5) and the β-smoothness assumption; (b) uses inequalities (B.7), (B.8). With probability $\frac{1}{n}$ the selected example is different, i.e., $j_t(i_t) = j_*$. With the bounded gradient assumption, we have

$$
\begin{aligned}
&\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| = \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_*}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}'_{j_*})\| \\
&\leq \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_*}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_*})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}_t; \mathbf{z}_{j_*}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_*}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}; \mathbf{z}_{j_*}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_*})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}'_t; \mathbf{z}_{j_*}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_*}) + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}; \mathbf{z}_{j_*}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_*})\| \\
&\quad + \|\frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_*}) - \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}'_{j_*})\| \\
&\leq (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}_{t-\tau_t}(i_t)\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_t - \mathbf{x}'_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}(i_t)\| \\
&\quad + \frac{\alpha_t}{m}\|\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}_{j_*}) - \nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}'_{j_*})\| \\
&\leq (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2\beta\alpha_t}{m}\frac{L}{m}\sum_{s=t-\tau_t}^{t-1}\alpha_s + \frac{2\beta\alpha_t}{m}L\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \frac{2L\alpha_t}{m} \\
&\leq (1 - \frac{\mu\alpha_t}{m})\|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2L\alpha_t}{m} + \frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big).
\end{aligned}
\tag{B.17}
$$

Combining the inequalities (B.16) and (B.17), we have

$$
\begin{aligned}
\mathbb{E}[\delta_{t+1}] &\leq (1 - \frac{1}{n})(1 - \frac{\mu\alpha_t}{m})\mathbb{E}[\delta_t] + (1 - \frac{1}{n})\frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big) \\
&\quad + \frac{1}{n}(1 - \frac{\mu\alpha_t}{m})\mathbb{E}[\delta_t] + \frac{1}{n}\frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big) + \frac{1}{n}\frac{2L\alpha_t}{m} \\
&\leq (1 - \frac{\mu\alpha_t}{m})\mathbb{E}[\delta_t] + \frac{2L\alpha_t}{nm} + \frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big).
\end{aligned}
$$

We then derive

$$
\mathbb{E}[\delta_T] \leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1 - \frac{\mu\alpha_k}{m})\Big)\Big[\frac{2L\alpha_t}{nm} + \frac{2\beta L\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big]. \tag{B.18}
$$

For every $\mathbf{z}$, the L-Lipschitz condition indicate that

$$
\mathbb{E}|f(\mathbf{x}_T; \mathbf{z}) - f(\mathbf{x}'_T; \mathbf{z})| \leq L\mathbb{E}[\delta_T] \leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1 - \frac{\mu\alpha_k}{m})\Big)\cdot\Big[\frac{2L^2\alpha_t}{nm} + \frac{2\beta L^2\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big],
$$

which means the uniform stability satisfies

$$
\epsilon_{\text{stab}} \leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1 - \frac{\mu\alpha_k}{m})\Big)\Big[\frac{2L^2\alpha_t}{nm} + \frac{2\beta L^2\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big].
$$

For the constant learning rate $\alpha_t = \alpha$, we have

$$\begin{aligned}
\epsilon_{\text{stab}} &\leq \sum_{t=1}^{T-1}\Big((1-\frac{\mu\alpha}{m})^{T-1-t}\Big)\Big[\frac{2L^2\alpha}{nm} + \frac{2\beta L^2\alpha^2}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{1}{m}\Big)\Big] \\
&\leq \Big[\frac{2L^2\alpha}{nm} + \frac{2\beta L^2\alpha^2}{m}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big)\Big]\cdot\sum_{t=1}^{T-1}(1-\frac{\mu\alpha}{m})^{T-1-t} \\
&\leq \Big[\frac{2L^2\alpha}{nm} + \frac{2\beta L^2\alpha^2}{m}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big)\Big]\cdot\frac{m}{\mu\alpha} \\
&\leq \frac{2L^2}{\mu n} + \frac{2\beta L^2\alpha}{\mu}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big).
\end{aligned}$$

For the decreasing learning rate $\alpha_t = \frac{m}{\mu(t+1)}$, the stability turns to

$$\begin{aligned}
\epsilon_{\text{stab}} &\leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1-\frac{1}{k+1})\Big)\Big[\frac{2L^2}{\mu n(t+1)} + \frac{2\beta L^2}{\mu(t+1)}\Big(\frac{m}{\mu}\sum_{s=1}^{t-\tau_t-1}\frac{\lambda^{t-\tau_t-1-s}}{s+1} + \frac{1}{\mu}\sum_{s=t-\tau_t}^{t-1}\frac{1}{s+1}\Big)\Big] \\
&\leq \sum_{t=1}^{T-1}\frac{t+1}{T}\Big[\frac{2L^2}{\mu n(t+1)} + \frac{2\beta L^2}{\mu(t+1)}\Big(\frac{m}{\mu\lambda^{\overline{\tau}}}\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} + \frac{1}{\mu}\sum_{s=t-\tau_t}^{t-1}\frac{1}{s+1}\Big)\Big] \\
&\overset{(a)}{\leq} \sum_{t=1}^{T-1}\frac{t+1}{T}\Big[\frac{2L^2}{\mu n(t+1)} + \frac{2\beta L^2}{\mu(t+1)}\Big(\frac{mC_\lambda}{\mu t\lambda^{\overline{\tau}}} + \frac{\tau_t}{\mu(t-\tau_t+1)}\Big)\Big] \\
&\leq \sum_{t=1}^{T-1}\Big[\frac{2L^2}{\mu nT} + \frac{2\beta L^2}{\mu T}\Big(\frac{mC_\lambda}{\mu t\lambda^{\overline{\tau}}} + \frac{\overline{\tau}}{\mu(t-\tau_t+1)}\Big)\Big] \\
&\overset{(b)}{\leq} \frac{2L^2}{\mu n} + \frac{2m\beta L^2C_\lambda}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T+1}{T} + \frac{2\beta L^2}{\mu^2}\frac{\overline{\tau}^2+\overline{\tau}\ln T}{T} \\
&\leq \frac{2L^2}{\mu n} + \frac{2\beta L^2(mC_\lambda+\overline{\tau}^2\lambda^{\overline{\tau}})}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T+1}{T},
\end{aligned}$$

where (a) uses the inequality (B.6), and (b) uses the following inequalities

$$\sum_{t=1}^{T-1}\frac{1}{t} = 1 + \sum_{t=1}^{T-2}\frac{1}{t+1} \leq 1 + \sum_{t=1}^{T-2}\int_t^{t+1}\frac{1}{x}dx \leq 1 + \int_1^{T-1}\frac{1}{x}dx \leq \ln T + 1; \tag{B.19}$$

$$\sum_{t=1}^{T-1}\frac{1}{t-\tau_t+1} \leq \sum_{t=1}^{\overline{\tau}}\frac{1}{t-\tau_t+1} + \sum_{t=\overline{\tau}+1}^{T-1}\frac{1}{t-\overline{\tau}+1} \leq \overline{\tau} + \sum_{t=1}^{T-\overline{\tau}-1}\frac{1}{t+1} \leq \overline{\tau} + \ln(T-\overline{\tau}) \leq \overline{\tau} + \ln T. \tag{B.20}$$

■

B.7 Proof of Theorem 3 (generalization error in the non-convex case)

$\mathbf{x}_T$ and $\mathbf{x}'_T$ denote the output model of running AD-SGD on $\mathcal{S}$ and $\mathcal{S}'$ for T iterations, respectively. With probability $1-\frac{1}{n}$, the example selected in $\mathcal{S}$ and $\mathcal{S}'$ is the same at the t-th iteration, i.e., $j_t(i_t) \neq j_*$. Then we have

$$\begin{aligned}
&\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| = \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\| \\
&\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\alpha_t}{m}\|\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}'_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\| \\
&\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\alpha_t}{m}\Big[\|\nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}'_{t-\tau_t};\mathbf{z}_{j_t(i_t)})\| \\
&\quad + \|\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)})\| + \|\nabla f(\mathbf{x}'_{t-\tau_t};\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}'_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\|\Big] \\
&\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}_{t-\tau_t}(i_t)\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}'_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}(i_t)\| \\
&\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{\beta\alpha_t}{m}\|\mathbf{x}_{t-\tau_t} - \mathbf{x}'_{t-\tau_t}\| + \frac{2\beta L\alpha_t}{m}\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s}.
\end{aligned} \tag{B.21}$$

With probability $\frac{1}{n}$, $j_t = j_*$, we can get

$$\begin{aligned}\|\mathbf{x}_{t+1} - \mathbf{x}'_{t+1}\| &= \|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_*}) - \mathbf{x}'_t + \frac{\alpha_t}{m}\nabla f(\mathbf{x}'_{t-\tau_t}(i_t); \mathbf{z}'_{j_*})\| \\ &\leq \|\mathbf{x}_t - \mathbf{x}'_t\| + \frac{2L\alpha_t}{m}.\end{aligned} \tag{B.22}$$

Combining inequalities (B.21) and (B.22), we have

$$\begin{aligned}\mathbb{E}[\delta_{t+1}] &\leq (1-\frac{1}{n})\mathbb{E}[\delta_t] + (1-\frac{1}{n})\frac{\beta\alpha_t}{m}\mathbb{E}[\delta_{t-\tau_t}] + (1-\frac{1}{n})\frac{2\beta L\alpha_t}{m}\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \frac{1}{n}\mathbb{E}[\delta_t] + \frac{1}{n}\frac{2L\alpha_t}{m} \\ &\leq \mathbb{E}[\delta_t] + \frac{(n-1)\beta\alpha_t}{nm}\mathbb{E}[\delta_{t-\tau_t}] + \frac{2L\alpha_t}{nm} + \frac{2(n-1)\beta L\alpha_t}{nm}\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} \\ &\leq \mathbb{E}[\delta_t] + \frac{\beta\alpha_t}{m}\max_{t-\tau_t\leq k\leq t}\mathbb{E}[\delta_k] + \frac{2L\alpha_t}{m}\Big(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1}\beta\alpha_s.\lambda^{t-\tau_t-1-s}\Big).\end{aligned} \tag{B.23}$$

Following [Proposition 2, (Regatti et al. 2019)] and we define $\prod_{k=t'+1}^{t'}(1+\frac{\beta\alpha_k}{m}) = 1$. Then we have

$$\mathbb{E}[\delta_T] \leq \sum_{t=1}^{T-1}\left(\prod_{k=t+1}^{T-1}(1+\frac{\beta\alpha_k}{m})\right)\frac{2L\alpha_t}{m}\left(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1}\beta\alpha_s\lambda^{t-\tau_t-1-s}\right). \tag{B.24}$$

For every $\mathbf{z}$, the L-Lipschitz condition indicate that

$$\mathbb{E}|f(\mathbf{x}_T;\mathbf{z}) - f(\mathbf{x}'_T;\mathbf{z})| \leq L\mathbb{E}[\delta_T] \leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1+\frac{\beta\alpha_k}{m})\Big)\frac{2L^2\alpha_t}{m}\left(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1}\beta\alpha_s\lambda^{t-\tau_t-1-s}\right).$$

which means the uniform stability in the non-convex case satisfies

$$\epsilon_{\text{stab}} \leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1+\frac{\beta\alpha_k}{m})\Big)\frac{2L^2\alpha_t}{m}\left(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1}\beta\alpha_s\lambda^{t-\tau_t-1-s}\right). \tag{B.25}$$

■

B.8 Proof of Corollary 2 (generalization error for different learning rate in the non-convex case)

According to (B.25), for the constant learning rate $\alpha_t = \alpha$, we have

$$\begin{aligned}\epsilon_{\text{stab}} &\leq \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1+\frac{\beta\alpha}{m})\Big)\frac{2L^2\alpha}{m}\left(\frac{1}{n} + \sum_{s=1}^{t-\tau_t-1}\beta\alpha\lambda^{t-\tau_t-1-s}\right) \\ &\leq \Big(\frac{2L^2\alpha}{nm} + \frac{2\beta L^2\alpha^2}{m(1-\lambda)}\Big)\sum_{t=1}^{T-1}(1+\frac{\beta\alpha}{m})^{T-1-t} \\ &\leq \Big(\frac{2L^2\alpha}{nm} + \frac{2\beta L^2\alpha^2}{m(1-\lambda)}\Big)\frac{m}{\beta\alpha}\Big[(1+\frac{\beta\alpha}{m})^{T-1} - 1\Big] \\ &\leq \frac{2L^2(1+\beta n\alpha-\lambda)}{\beta n(1-\lambda)}(1+\frac{\beta\alpha}{m})^{T-1}.\end{aligned} \tag{B.26}$$

For the decreasing learning rate $\alpha_t = \frac{mc}{t+1}$, it follows that

$$\begin{aligned}
\epsilon_{\text{stab}} &\leq \sum_{t=1}^{T-1}\left\{\prod_{k=t+1}^{T-1}(1+\frac{\beta c}{k+1})\right\}\Big(\frac{2L^2c}{n(t+1)}+\frac{2\beta L^2mc^2}{t+1}\sum_{s=1}^{t-\tau_t-1}\frac{\lambda^{t-\tau_t-1-s}}{s+1}\Big)\\
&\overset{(a)}{\leq} \sum_{t=1}^{T-1}\left\{\prod_{k=t+1}^{T-1}\exp\Big(\frac{\beta c}{k+1}\Big)\right\}\Big(\frac{2L^2c}{n(t+1)}+\frac{2\beta L^2mc^2}{t+1}\sum_{s=1}^{t-\tau_t-1}\lambda^{t-\tau_t-1-s}\Big)\\
&\leq \sum_{t=1}^{T-1}\exp\Big(\beta c\sum_{k=t+1}^{T-1}\frac{1}{k+1}\Big)\Big[\frac{2L^2c}{n(t+1)}+\frac{2\beta L^2mc^2}{(1-\lambda)(t+1)}\Big]\\
&\overset{(b)}{\leq} \sum_{t=1}^{T-1}\exp\Big(\beta c\ln(\frac{T}{t+1})\Big)\Big[\frac{2L^2c}{n(t+1)}+\frac{2\beta L^2mc^2}{(1-\lambda)(t+1)}\Big]\\
&\leq \Big[\frac{2L^2c}{n}+\frac{2\beta L^2mc^2}{1-\lambda}\Big]T^{\beta c}\sum_{t=1}^{T-1}(t+1)^{-\beta c-1}\\
&\overset{(c)}{\leq} \Big[\frac{2L^2c}{n}+\frac{2\beta L^2mc^2}{1-\lambda}\Big]T^{\beta c}\frac{1}{\beta c}(1-\frac{1}{T^{\beta c}})\\
&\leq \frac{2L^2(1+\beta nmc-\lambda)}{\beta n(1-\lambda)}T^{\beta c},
\end{aligned} \tag{B.27}$$

where (a) uses $1+x\leq e^x$. (b) and (c) respectively use the following inequalities

$$\sum_{k=t+1}^{T-1}\frac{1}{k+1}\leq\sum_{k=t+1}^{T-1}\int_k^{k+1}\frac{1}{x}dx\leq\int_{t+1}^{T}\frac{1}{x}dx=\ln(\frac{T}{t+1});$$

$$\sum_{t=1}^{T-1}(t+1)^{-\beta c-1}\leq\sum_{t=1}^{T-1}\int_t^{t+1}x^{-\beta c-1}dx\leq\int_1^T x^{-\beta c-1}dx=\frac{1}{\beta c}(1-T^{-\beta c}).$$

With $c=1/\beta$, we have

$$\epsilon_{\text{stab}}\leq\frac{2L^2(1+nm-\lambda)}{\beta n(1-\lambda)}T.$$

■

B.9 Proof of Theorem 4 (generalization error for decreasing learning rate in the non-convex case)

Following [Lemma 3.11, (Hardt, Recht, and Singer 2016)], let $\delta_{t_0=0}$ and we have

$$\epsilon_{\text{stab}}\leq\frac{t_0}{n}+L\mathbb{E}[\delta_T|\delta_{t_0=0}].$$

Similar to the derivation in (B.27), we have

$$\mathbb{E}[\delta_T|\delta_{t_0=0}]\leq\frac{2L(1+\beta nmc-\lambda)}{\beta n(1-\lambda)}\left(\frac{T}{t_0}\right)^{\beta c}.$$

Then we get

$$\epsilon_{\text{stab}}\leq\frac{t_0}{n}+\frac{2L^2(1+\beta nmc-\lambda)}{\beta n(1-\lambda)}\left(\frac{T}{t_0}\right)^{\beta c}.$$

Assume c is small enough, minimizing this bound with respect to t_0, i.e., let

$$t_0=\Big[2L^2c(1+\frac{\beta nmc}{1-\lambda})\Big]^{\frac{1}{\beta c+1}}T^{\frac{\beta c}{\beta c+1}},$$

then the uniform stability satisfies

$$\epsilon_{\text{stab}}\leq\frac{1+1/\beta c}{n}\Big[2L^2c(1+\frac{\beta nmc}{1-\lambda})\Big]^{\frac{1}{\beta c+1}}T^{\frac{\beta c}{\beta c+1}}.$$

■

B.10 Proof of Theorem 5 (optimization error and excess generalization error in the strongly convex case)

Recall that $\mathbf{x}_t$ is the output model after minimizing the empirical risk $F_{\mathcal{S}}$ for t AD-SGD iterations, and $\mathbf{x}_{\mathcal{S}}^*$ denotes the minimizer of $F_{\mathcal{S}}$. From the iterative relation (B.2), we can derive

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{x}_{t+1}-\mathbf{x}_{\mathcal{S}}^*\|^2 = \mathbb{E}\|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) - \mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\le \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{\alpha_t^2}{m^2}\mathbb{E}\|\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\|^2 \\
&\le \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{2\alpha_t}{m}\mathbb{E}\langle \nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{L^2\alpha_t^2}{m^2} \\
&\overset{(a)}{\le} \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4r\alpha_t}{m}\mathbb{E}\|\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\| + \frac{L^2\alpha_t^2}{m^2} \\
&\le \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{L^2\alpha_t^2}{m^2} \\
&\quad + \frac{4r\alpha_t}{m}\left[\mathbb{E}\|\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)})\| + \mathbb{E}\|\nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\|\right] \\
&\le \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4\beta r\alpha_t}{m}\left[\|\mathbf{x}_t-\mathbf{x}_{t-\tau_t}\| + \|\mathbf{x}_{t-\tau_t}-\mathbf{x}_{t-\tau_t}(i_t)\|\right] + \frac{L^2\alpha_t^2}{m^2} \\
&\overset{(b)}{\le} \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4\beta rL\alpha_t}{m}\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{L^2\alpha_t^2}{m^2} \\
&\overset{(c)}{\le} (1-\frac{2\mu\alpha_t}{m})\mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{4\beta rL\alpha_t}{m}\left(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right) + \frac{L^2\alpha_t^2}{m^2},
\end{aligned}
\tag{B.28}
$$

where (a) uses the inequality $\langle \mathbf{a},\mathbf{b}\rangle \le \|\mathbf{a}\|\|\mathbf{b}\|$ and Assumption 4 (r is the radius of the close ball). (b) uses inequalities (B.7) and (B.8). (c) employs the following μ-strongly convexity

$$
\langle \nabla f(\mathbf{x}_t), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle \ge \mu\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2.
$$

We then have

$$
\begin{aligned}
\mathbb{E}\|\mathbf{x}_T-\mathbf{x}_{\mathcal{S}}^*\|^2 \le &\sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1-\frac{2\mu\alpha_k}{m})\Big)\Big[\frac{L^2\alpha_t^2}{m^2} + \frac{4\beta rL\alpha_t}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\Big)\Big] \\
&+ \prod_{t=1}^{T-1}(1-\frac{2\mu\alpha_t}{m})\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2.
\end{aligned}
$$

For the constant learning rate $\alpha_t = \alpha$

$$
\begin{aligned}
\mathbb{E}\|\mathbf{x}_T-\mathbf{x}_{\mathcal{S}}^*\|^2 &\le \sum_{t=1}^{T-1}\Big((1-\frac{2\mu\alpha}{m})^{T-1-t}\Big)\Big[\frac{L^2\alpha^2}{m^2} + \frac{4\beta rL\alpha^2}{m}\Big(\sum_{s=1}^{t-\tau_t-1}\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{1}{m}\Big)\Big] + (1-\frac{2\mu\alpha}{m})^{T-1}\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\le \Big[\frac{L^2\alpha^2}{m^2} + \frac{4\beta rL\alpha^2}{m}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big)\Big]\cdot\sum_{t=1}^{T-1}(1-\frac{2\mu\alpha}{m})^{T-1-t} + (1-\frac{2\mu\alpha}{m})^{T-1}\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\le \Big[\frac{L^2\alpha^2}{m^2} + \frac{4\beta rL\alpha^2}{m}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big)\Big]\cdot\frac{m}{2\mu\alpha} + (1-\frac{2\mu\alpha}{m})^{T-1}\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\le \frac{L^2\alpha}{2\mu m} + \frac{2\beta rL\alpha}{\mu}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big) + (1-\frac{2\mu\alpha}{m})^{T-1}\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2.
\end{aligned}
$$

With β-smooth property, the optimization error satisfies

$$
\begin{aligned}
\epsilon_{\text{opt}} = \mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_T) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)] &\le \mathbb{E}\langle \nabla F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*), \mathbf{x}_T-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{\beta}{2}\mathbb{E}\|\mathbf{x}_T-\mathbf{x}_{\mathcal{S}}^*\|^2 \le \frac{\beta}{2}\mathbb{E}\|\mathbf{x}_T-\mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\le \frac{\beta L^2\alpha}{4\mu m} + \frac{\beta^2 rL\alpha}{\mu}\Big(\frac{1}{1-\lambda}+\frac{\overline{\tau}}{m}\Big) + (1-\frac{2\mu\alpha}{m})^{T-1}\frac{\beta\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2}{2}.
\end{aligned}
$$

Following the decomposition (1), the excess generalization error satisfies

$$\begin{aligned}\epsilon_{\text{exc}} &\le \epsilon_{\text{stab}} + \epsilon_{\text{opt}} \\ &\le \frac{2L^2}{\mu n} + \frac{2\beta L^2\alpha}{\mu}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big) + \frac{\beta L^2\alpha}{4\mu m} + \frac{\beta^2 rL\alpha}{\mu}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big) + (1 - \frac{2\mu\alpha}{m})^{T-1}\frac{\beta\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2} \\ &\le \frac{L^2(8m + \beta n\alpha)}{4\mu n m} + \frac{\beta L\alpha(2L + \beta r)}{\mu}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big) + (1 - \frac{2\mu\alpha}{m})^{T-1}\frac{\beta\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2}.\end{aligned}$$

For the decreasing learning rate $\alpha_t = \frac{m}{2\mu(t+1)}$, we have

$$\begin{aligned}&\mathbb{E}\|\mathbf{x}_T - \mathbf{x}_{\mathcal{S}}^*\|^2 \\ &\le \sum_{t=1}^{T-1}\Big(\prod_{k=t+1}^{T-1}(1 - \frac{1}{k+1})\Big)\Big[\frac{L^2}{4\mu^2(t+1)^2} + \frac{2\beta rL}{\mu(t+1)}\Big(\frac{m}{2\mu}\sum_{s=1}^{t-\tau_t-1}\frac{\lambda^{t-\tau_t-1-s}}{s+1} + \frac{1}{2\mu}\sum_{s=t-\tau_t}^{t-1}\frac{1}{s+1}\Big)\Big] \\ &\quad + \prod_{t=1}^{T-1}(1 - \frac{1}{t+1})\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2 \\ &\le \sum_{t=1}^{T-1}\frac{t+1}{T}\Big[\frac{L^2}{4\mu^2(t+1)^2} + \frac{2\beta rL}{\mu(t+1)}\Big(\frac{m}{2\mu\lambda^{\overline{\tau}}}\sum_{s=1}^{t-1}\frac{\lambda^{t-1-s}}{s+1} + \frac{1}{2\mu}\sum_{s=t-\tau_t}^{t-1}\frac{1}{s+1}\Big)\Big] + \frac{\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{T} \\ &\overset{(a)}{\le} \sum_{t=1}^{T-1}\frac{t+1}{T}\Big[\frac{L^2}{4\mu^2(t+1)^2} + \frac{2\beta rL}{\mu(t+1)}\Big(\frac{mC_\lambda}{2\mu t\lambda^{\overline{\tau}}} + \frac{\tau_t}{2\mu(t-\tau_t+1)}\Big)\Big] + \frac{\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{T} \\ &\le \sum_{t=1}^{T-1}\Big[\frac{L^2}{4\mu^2 T(t+1)} + \frac{2\beta rL}{\mu T}\Big(\frac{mC_\lambda}{2\mu t\lambda^{\overline{\tau}}} + \frac{\overline{\tau}}{2\mu(t-\tau_t+1)}\Big)\Big] + \frac{\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{T} \\ &\overset{(b)}{\le} \frac{L^2\ln T}{4\mu^2 T} + \frac{\beta rLmC_\lambda}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\beta rL}{\mu^2}\frac{\overline{\tau}^2 + \overline{\tau}\ln T}{T} + \frac{\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{T} \\ &\le \frac{L^2\ln T}{4\mu^2 T} + \frac{\beta rL(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{T},\end{aligned}$$

where (a) uses inequality (B.6), and (b) uses (B.14), (B.19) and (B.20). With β-smooth property, the optimization error satisfies

$$\begin{aligned}\epsilon_{\text{opt}} = \mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_T) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)] &\le \mathbb{E}\langle\nabla F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*), \mathbf{x}_T - \mathbf{x}_{\mathcal{S}}^*\rangle + \frac{\beta}{2}\mathbb{E}\|\mathbf{x}_T - \mathbf{x}_{\mathcal{S}}^*\|^2 \le \frac{\beta}{2}\mathbb{E}\|\mathbf{x}_T - \mathbf{x}_{\mathcal{S}}^*\|^2 \\ &\le \frac{\beta L^2\ln T}{8\mu^2 T} + \frac{\beta^2 rL(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{2\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\beta\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2T} \\ &\le \frac{\beta L^2\ln T}{8\mu^2 T} + \frac{\beta^2 rL(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{2\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{2\beta r^2}{T}.\end{aligned}$$

Following the decomposition (1), the excess generalization risk satisfies

$$\begin{aligned}\epsilon_{\text{exc}} &\le \epsilon_{\text{stab}} + \epsilon_{\text{opt}} \\ &\le \frac{2L^2}{\mu n} + \frac{2\beta L^2(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\beta L^2\ln T}{8\mu^2 T} + \frac{\beta^2 rL(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{2\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\beta\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2T} \\ &\le \frac{2L^2}{\mu n} + \frac{\beta L(4L + \beta r)(mC_\lambda + \overline{\tau}^2\lambda^{\overline{\tau}})}{2\mu^2\lambda^{\overline{\tau}}}\frac{\ln T + 1}{T} + \frac{\beta L^2\ln T}{8\mu^2 T} + \frac{\beta\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2T}.\end{aligned}$$

■

B.11 Proof of Theorem 6 and 7 (optimization error and excess generalization error in the convex case)

Similar to the analysis in (B.28), we have the following relationship

$$
\begin{aligned}
&\mathbb{E}\|\mathbf{x}_{t+1}-\mathbf{x}_{\mathcal{S}}^*\|^2 = \mathbb{E}\|\mathbf{x}_t - \frac{\alpha_t}{m}\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}) - \mathbf{x}_{\mathcal{S}}^*\|^2 \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{\alpha_t^2}{m^2}\mathbb{E}\|\nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\|^2 \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{2\alpha_t}{m}\mathbb{E}\langle \nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{L^2\alpha_t^2}{m^2} \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4r\alpha_t}{m}\mathbb{E}\|\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\| + \frac{L^2\alpha_t^2}{m^2} \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{L^2\alpha_t^2}{m^2} \\
&\quad + \frac{4r\alpha_t}{m}\left[\mathbb{E}\|\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)})\| + \mathbb{E}\|\nabla f(\mathbf{x}_{t-\tau_t};\mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t);\mathbf{z}_{j_t(i_t)})\|\right] \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4\beta r\alpha_t}{m}\left[\|\mathbf{x}_t-\mathbf{x}_{t-\tau_t}\| + \|\mathbf{x}_{t-\tau_t}-\mathbf{x}_{t-\tau_t}(i_t)\|\right] + \frac{L^2\alpha_t^2}{m^2} \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 + \frac{2\alpha_t}{m}\mathbb{E}\langle -\nabla f(\mathbf{x}_t;\mathbf{z}_{j_t(i_t)}), \mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\rangle + \frac{4\beta rL\alpha_t}{m}\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{L^2\alpha_t^2}{m^2} \\
&\leq \mathbb{E}\|\mathbf{x}_t-\mathbf{x}_{\mathcal{S}}^*\|^2 - \frac{2\alpha_t}{m}\mathbb{E}\left[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)\right] + \frac{4\beta rL\alpha_t}{m}\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{L^2\alpha_t^2}{m^2}.
\end{aligned}
$$

The last inequality uses the unbiased property of the stochastic gradient and the convexity of the loss function, i.e.,

$$\langle \nabla F_{\mathcal{S}}(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{\mathcal{S}}^*\rangle \geq F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*).$$

Then we have

$$\sum_{t=1}^T \alpha_t \mathbb{E}\left[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)\right] \leq \frac{m}{2}\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2 + 2\beta rL\sum_{t=1}^T\alpha_t\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{L^2}{2m}\sum_{t=1}^T\alpha_t^2.$$

Devote the average model as

$$\overline{\mathbf{x}}_T = \frac{\sum_{t=1}^T \alpha_t\mathbf{x}_t}{\sum_{t=1}^T\alpha_t}.$$

It follows that

$$
\begin{aligned}
\epsilon_{\text{opt}} = \mathbb{E}[F_{\mathcal{S}}(\overline{\mathbf{x}}_T) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)] &\leq \frac{\sum_{t=1}^T\alpha_t\mathbb{E}\left[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)\right]}{\sum_{t=1}^T\alpha_t} \\
&\leq \frac{m\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2}{2\sum_{t=1}^T\alpha_t} + \frac{2\beta rL}{\sum_{t=1}^T\alpha_t}\sum_{t=1}^T\alpha_t\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{L^2\sum_{t=1}^T\alpha_t^2}{2m\sum_{t=1}^T\alpha_t}.
\end{aligned}
$$

For the constant learning rate $\alpha_t = \alpha$

$$
\begin{aligned}
\epsilon_{\text{opt}} &\leq \frac{m\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2}{2\sum_{t=1}^T\alpha} + \frac{2\beta rL}{\sum_{t=1}^T\alpha}\sum_{t=1}^T\alpha\left[\sum_{s=1}^{t-\tau_t-1}\alpha\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha}{m}\right] + \frac{L^2\sum_{t=1}^T\alpha^2}{2m\sum_{t=1}^T\alpha} \\
&\leq \frac{m\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2}{2T\alpha} + \frac{2\beta rL}{T\alpha}\sum_{t=1}^T\alpha^2\left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{L^2T\alpha^2}{2mT\alpha} \\
&\leq \frac{m\|\mathbf{x}_1-\mathbf{x}_{\mathcal{S}}^*\|^2}{2T\alpha} + 2\beta rL\alpha\left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{L^2\alpha}{2m}.
\end{aligned}
$$

For the decreasing learning rate $\alpha_t = \frac{1}{t+1}$, we have

$$\begin{aligned}
\epsilon_{\text{opt}} &\leq \frac{m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2\sum_{t=1}^T \frac{1}{t+1}} + \frac{2\beta r L}{\sum_{t=1}^T \frac{1}{t+1}} \sum_{t=1}^T \frac{1}{t+1}\left[\sum_{s=1}^{t-\tau_t-1} \frac{\lambda^{t-\tau_t-1-s}}{s+1} + \sum_{s=t-\tau_t}^{t-1} \frac{1}{m(s+1)}\right] + \frac{L^2 \sum_{t=1}^T \frac{1}{(t+1)^2}}{2m\sum_{t=1}^T \frac{1}{t+1}} \\
&\overset{(a)}{\leq} \frac{m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{\ln(T+1)} + \frac{4\beta r L}{\ln(T+1)} \sum_{t=1}^T \frac{1}{t+1}\left[\frac{C_\lambda}{t\lambda^{\overline{\tau}}} + \frac{\tau_t}{m(t-\tau_t+1)}\right] + \frac{L^2}{m\ln(T+1)} \\
&\overset{(b)}{\leq} \frac{m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{\ln(T+1)} + \frac{4\beta r L}{\ln(T+1)}\Big[\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{\overline{\tau}+\ln(\overline{\tau}+1)}{m}\Big] + \frac{L^2}{m\ln(T+1)} \\
&\leq \Big[m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2 + 4\beta r L(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{L^2}{m}\Big]\frac{1}{\ln(T+1)} \\
&\leq \Big[4mr^2 + 4\beta r L(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{L^2}{m}\Big]\frac{1}{\ln(T+1)},
\end{aligned}$$

where (a) uses (B.6) and the following inequalities

$$\sum_{t=1}^T \frac{1}{t+1} \geq \frac{1}{2}\sum_{t=1}^T \frac{1}{t} \geq \frac{1}{2}\sum_{t=1}^T \int_t^{t+1} \frac{1}{x}dx \geq \frac{1}{2}\ln(T+1); \tag{B.29}$$

$$\sum_{t=1}^T \frac{1}{(t+1)^2} \leq \sum_{t=1}^T \int_t^{t+1} \frac{1}{x^2}dx \leq \int_1^{T+1} \frac{1}{x^2}dx \leq 1 - \frac{1}{T+1} \leq 1. \tag{B.30}$$

(b) uses inequality (B.15). In the following, we fist derive the uniform stability bound for the average model $\overline{\mathbf{x}}_T$. From the analysis in (B.12), we have

$$\mathbb{E}\|\mathbf{x}_t - \mathbf{x}_t'\| \leq \frac{2L}{n}\sum_{k=1}^{t-1}\frac{\alpha_k}{m} + 2\beta L\sum_{k=1}^{t-1}\frac{\alpha_k}{m}\Big[\sum_{s=1}^{k-\tau_k-1}\alpha_s\lambda^{k-\tau_k-1-s} + \sum_{s=k-\tau_k}^{k-1}\frac{\alpha_s}{m}\Big].$$

Then we can derive

$$\begin{aligned}
\mathbb{E}\|\overline{\mathbf{x}}_T - \overline{\mathbf{x}}_T'\| &= \mathbb{E}\left\|\frac{\sum_{t=1}^T \alpha_t(\mathbf{x}_t - \mathbf{x}_t')}{\sum_{t=1}^T \alpha_t}\right\| \leq \frac{\sum_{t=1}^T \alpha_t \mathbb{E}\|\mathbf{x}_t - \mathbf{x}_t'\|}{\sum_{t=1}^T \alpha_t} \\
&\leq \frac{\frac{2L}{nm}\sum_{t=1}^T \alpha_t \sum_{k=1}^{t-1}\alpha_k + \frac{2\beta L}{m}\sum_{t=1}^T \alpha_t \sum_{k=1}^{t-1}\alpha_k\Big[\sum_{s=1}^{k-\tau_k-1}\alpha_s\lambda^{k-\tau_k-1-s} + \sum_{s=k-\tau_k}^{k-1}\frac{\alpha_s}{m}\Big]}{\sum_{t=1}^T \alpha_t}.
\end{aligned}$$

For the constant learning rate $\alpha_t = \alpha$

$$\begin{aligned}
\mathbb{E}\|\overline{\mathbf{x}}_T - \overline{\mathbf{x}}_T'\| &\leq \frac{\frac{2L\alpha^2}{nm}\sum_{t=1}^T (t-1) + \frac{2\beta L\alpha^3}{m}\sum_{t=1}^T (t-1)\Big[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big]}{T\alpha} \\
&\leq \frac{\frac{2L\alpha^2}{nm}\frac{T(T-1)}{2} + \frac{2\beta L\alpha^3}{m}\frac{T(T-1)}{2}\Big[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big]}{T\alpha} \\
&\leq \frac{L\alpha(T-1)}{nm} + \frac{\beta L\alpha^2(T-1)}{m}\Big[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big].
\end{aligned}$$

Combine with the L-Lipschitz condition, the uniform stability bound of $\overline{\mathbf{x}}_T$ satisfies

$$\epsilon_{\text{ave-stab}} \leq \frac{L^2\alpha(T-1)}{nm} + \frac{\beta L^2\alpha^2(T-1)}{m}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big).$$

Then the excess generalization risk follows

$$\begin{aligned}
\epsilon_{\text{exc}} &\leq \epsilon_{\text{ave-stab}} + \epsilon_{\text{opt}} \\
&\leq \frac{L^2\alpha(T-1)}{nm} + \frac{\beta L^2\alpha^2(T-1)}{m}\Big(\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\Big) + \frac{m\|\mathbf{x}_1 - \mathbf{x}_{\mathcal{S}}^*\|^2}{2T\alpha} + 2\beta r L\alpha\left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{L^2\alpha}{2m}.
\end{aligned}$$

For the decreasing learning rate $\alpha_t = \frac{1}{t+1}$

$$\begin{aligned}
\mathbb{E}\|\overline{\mathbf{x}}_T - \overline{\mathbf{x}}'_T\| &\leq \frac{\frac{2L}{nm}\sum_{t=1}^{T}\frac{1}{t+1}\sum_{k=1}^{t-1}\frac{1}{k+1} + \frac{2\beta L}{m}\sum_{t=1}^{T}\frac{1}{t+1}\sum_{k=1}^{t-1}\frac{1}{k+1}\left[\sum_{s=1}^{k-\tau_k-1}\frac{1}{s+1}\lambda^{k-\tau_k-1-s} + \sum_{s=k-\tau_k}^{k-1}\frac{1}{m(s+1)}\right]}{\sum_{t=1}^{T}\frac{1}{t+1}} \\
&\overset{(a)}{\leq} \frac{\frac{4L}{nm}\sum_{t=1}^{T}\frac{\ln t}{t+1} + \frac{4\beta L}{m}\sum_{t=1}^{T}\frac{1}{t+1}\sum_{k=1}^{t-1}\frac{1}{k+1}\left[\frac{C_\lambda}{\lambda^{\overline{\tau}}}\frac{1}{k} + \frac{\tau_k}{m(k-\tau_k-1)}\right]}{\ln(T+1)} \\
&\overset{(b)}{\leq} \frac{\frac{2L}{nm}\ln^2(T+1) + \frac{4\beta L}{m}\sum_{t=1}^{T}\frac{1}{t+1}\left[\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{\overline{\tau}+\ln(\overline{\tau}+1)}{m}\right]}{\ln(T+1)} \\
&\leq \frac{2L}{nm}\ln(T+1) + \frac{4\beta L}{m}\left(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\right),
\end{aligned}$$

where (a) uses inequalities (B.6), (B.14) and (B.29). (b) uses inequality (B.20) and

$$\sum_{t=1}^{T}\frac{\ln t}{t+1} \leq \sum_{t=1}^{T}\int_{t}^{t+1}\frac{\ln x}{x}dx \leq \int_{1}^{T+1}\frac{\ln x}{x}dx = \frac{\ln^2(T+1)}{2}.$$

Then the uniform stability bound of $\overline{\mathbf{x}}_T$ satisfies

$$\epsilon_{\text{ave-stab}} \leq \frac{2L^2}{nm}\ln(T+1) + \frac{4\beta L^2}{m}\left(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\right).$$

The excess generalization risk in the decreasing learning rate follows

$$\begin{aligned}
\epsilon_{\text{exc}} &\leq \epsilon_{\text{ave-stab}} + \epsilon_{\text{opt}} \\
&\leq \frac{2L^2}{nm}\ln(T+1) + \frac{4\beta L^2}{m}\left(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}\right) + \left[m\|\mathbf{x}_1 - \mathbf{x}^*_{\mathcal{S}}\|^2 + 4\beta r L(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{L^2}{m}\right]\frac{1}{\ln(T+1)}.
\end{aligned}$$

■

B.12 Proof of Theorem 8 and 9 (optimization error and excess generalization error in the non-convex case)

With the β-smooth property, we have

$$\begin{aligned}
&\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_{t+1}) - F_{\mathcal{S}}(\mathbf{x}_t)] \leq \mathbb{E}\langle\nabla F_{\mathcal{S}}(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t\rangle + \frac{\beta}{2}\mathbb{E}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\
&\leq \mathbb{E}\langle\nabla F_{\mathcal{S}}(\mathbf{x}_t), -\frac{\alpha_t}{m}\nabla F_{\mathcal{S}}(\mathbf{x}_t)\rangle + \mathbb{E}\left\langle\nabla f(\mathbf{x}_t(i_t); \mathbf{z}_{j_t}), \frac{\alpha_t}{m}\left(\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\right)\right\rangle \\
&\quad + \frac{\beta\alpha_t^2}{2m^2}\mathbb{E}\|\nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\|^2 \\
&\leq -\frac{\alpha_t}{m}\mathbb{E}\|\nabla F_{\mathcal{S}}(\mathbf{x}_t)\|^2 + \frac{\alpha_t L}{m}\mathbb{E}\|\nabla f(\mathbf{x}_t; \mathbf{z}_{j_t(i_t)}) - \nabla f(\mathbf{x}_{t-\tau_t}(i_t); \mathbf{z}_{j_t(i_t)})\| + \frac{\beta L^2\alpha_t^2}{2m^2} \\
&\overset{(a)}{\leq} -\frac{2\gamma\alpha_t}{m}\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})] + \frac{\beta\alpha_t L}{m}\mathbb{E}\left[\|\mathbf{x}_t - \mathbf{x}_{t-\tau_t}\| + \|\mathbf{x}_{t-\tau_t} - \mathbf{x}_{t-\tau_t}(i_t)\|\right] + \frac{\beta L^2\alpha_t^2}{2m^2} \\
&\leq -\frac{2\gamma\alpha_t}{m}\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})] + \frac{\beta\alpha_t L^2}{m}\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{\beta L^2\alpha_t^2}{2m^2},
\end{aligned}$$

where (a) uses the following γ-PŁ condition

$$2\gamma[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})] \leq \|\nabla F_{\mathcal{S}}(\mathbf{x}_t)\|^2. \tag{B.31}$$

Then we have

$$\sum_{t=1}^{T}\alpha_t\mathbb{E}\left[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}^*_{\mathcal{S}})\right] \leq \frac{m}{2\gamma}\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_1) - F_{\mathcal{S}}(\mathbf{x}_{T+1})] + \frac{\beta L^2}{2\gamma}\sum_{t=1}^{T}\alpha_t\left[\sum_{s=1}^{t-\tau_t-1}\alpha_s\lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1}\frac{\alpha_s}{m}\right] + \frac{\beta L^2}{4\gamma m}\sum_{t=1}^{T}\alpha_t^2.$$

The optimization error satisfies

$$\begin{aligned}\epsilon_{\text{opt}} = \mathbb{E}[F_{\mathcal{S}}(\overline{\mathbf{x}}_T) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)] &\leq \frac{\sum_{t=1}^T \alpha_t \mathbb{E}\left[F_{\mathcal{S}}(\mathbf{x}_t) - F_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}^*)\right]}{\sum_{t=1}^T \alpha_t} \\ &\leq \frac{m\mathbb{E}[F_{\mathcal{S}}(\mathbf{x}_1) - F_{\mathcal{S}}(\mathbf{x}_{T+1})]}{2\gamma\sum_{t=1}^T \alpha_t} + \frac{\beta L^2}{2\gamma\sum_{t=1}^T \alpha_t}\sum_{t=1}^T \alpha_t \left[\sum_{s=1}^{t-\tau_t-1} \alpha_s \lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1} \frac{\alpha_s}{m}\right] + \frac{\beta L^2 \sum_{t=1}^T \alpha_t^2}{4\gamma m \sum_{t=1}^T \alpha_t} \\ &\leq \frac{Lmr}{\gamma\sum_{t=1}^T \alpha_t} + \frac{\beta L^2}{2\gamma\sum_{t=1}^T \alpha_t}\sum_{t=1}^T \alpha_t \left[\sum_{s=1}^{t-\tau_t-1} \alpha_s \lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1} \frac{\alpha_s}{m}\right] + \frac{\beta L^2 \sum_{t=1}^T \alpha_t^2}{4\gamma m \sum_{t=1}^T \alpha_t}.\end{aligned}$$

For the constant learning rate $\alpha_t = \alpha$

$$\begin{aligned}\epsilon_{\text{opt}} &\leq \frac{Lmr}{\gamma\sum_{t=1}^T \alpha} + \frac{\beta L^2}{2\gamma\sum_{t=1}^T \alpha}\sum_{t=1}^T \alpha \left[\sum_{s=1}^{t-\tau_t-1} \alpha \lambda^{t-\tau_t-1-s} + \sum_{s=t-\tau_t}^{t-1} \frac{\alpha}{m}\right] + \frac{\beta L^2 \sum_{t=1}^T \alpha^2}{4\gamma m \sum_{t=1}^T \alpha} \\ &\leq \frac{Lmr}{T\gamma\alpha} + \frac{\beta L^2}{2T\gamma\alpha}\sum_{t=1}^T \alpha^2 \left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{\beta L^2 T\alpha^2}{4\gamma m T\alpha} \\ &\leq \frac{Lmr}{T\gamma\alpha} + \frac{\beta L^2\alpha}{2\gamma}\left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{\beta L^2\alpha}{4\gamma m}.\end{aligned} \tag{B.32}$$

For the decreasing learning rate $\alpha_t = \frac{mc}{t+1}$, we have

$$\begin{aligned}\epsilon_{\text{opt}} &\leq \frac{Lmr}{\gamma\sum_{t=1}^T \frac{mc}{t+1}} + \frac{\beta L^2}{2\gamma\sum_{t=1}^T \frac{mc}{t+1}}\sum_{t=1}^T \frac{mc}{t+1}\left[mc\sum_{s=1}^{t-\tau_t-1} \frac{\lambda^{t-\tau_t-1-s}}{s+1} + c\sum_{s=t-\tau_t}^{t-1} \frac{1}{s+1}\right] + \frac{\beta L^2 \sum_{t=1}^T (\frac{mc}{t+1})^2}{4\gamma m \sum_{t=1}^T \frac{mc}{t+1}} \\ &\overset{(a)}{\leq} \frac{2Lr}{\gamma c\ln(T+1)} + \frac{\beta L^2 c}{\gamma\ln(T+1)}\sum_{t=1}^T \frac{1}{t+1}\left[\frac{mC_\lambda}{t\lambda^{\overline{\tau}}} + \frac{\tau_t}{t-\tau_t+1}\right] + \frac{\beta L^2 c}{2\gamma\ln(T+1)} \\ &\overset{(b)}{\leq} \frac{2Lr}{\gamma c\ln(T+1)} + \frac{\beta L^2 c}{\gamma\ln(T+1)}\left[\frac{mC_\lambda}{\lambda^{\overline{\tau}}} + \overline{\tau} + \ln(\overline{\tau}+1)\right] + \frac{\beta L^2 c}{2\gamma\ln(T+1)} \\ &\leq \left[2Lr + \beta m L^2 c^2(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{\beta L^2 c^2}{2}\right]\frac{1}{\gamma c\ln(T+1)},\end{aligned} \tag{B.33}$$

where (a) uses inequalities (B.6), (B.29) and (B.30). With $c = \frac{1}{\gamma}$, we then get

$$\epsilon_{\text{opt}} \leq \left[2Lr + \frac{\beta m L^2}{\gamma^2}(\frac{C_\lambda}{\lambda^{\overline{\tau}}} + \frac{2\overline{\tau}}{m}) + \frac{\beta L^2}{2\gamma^2}\right]\frac{1}{\ln(T+1)}.$$

For the constant learning rate $\alpha_t = \alpha$, it follows from (B.26) that

$$\begin{aligned}\mathbb{E}\|\overline{\mathbf{x}}_T - \overline{\mathbf{x}}_T'\| &\leq \frac{\frac{2L\alpha(1+\beta n\alpha-\lambda)}{\beta n(1-\lambda)}\sum_{t=1}^T (1+\frac{\beta\alpha}{m})^{t-1}}{T\alpha} \\ &\leq \frac{\frac{2L\alpha(1+\beta n\alpha-\lambda)}{\beta n(1-\lambda)}\frac{m}{\beta\alpha}(1+\frac{\beta\alpha}{m})^T}{T\alpha} \\ &\leq \frac{2Lm(1+\beta n\alpha-\lambda)}{\beta^2 n\alpha(1-\lambda)}\frac{(1+\frac{\beta\alpha}{m})^T}{T}.\end{aligned}$$

Then the uniform stability bound of $\overline{\mathbf{x}}_T$ satisfies

$$\epsilon_{\text{ave-stab}} \leq \frac{2L^2 m(1+\beta n\alpha-\lambda)}{\beta^2 n\alpha(1-\lambda)}\frac{(1+\frac{\beta\alpha}{m})^T}{T}.$$

Combined with the optimization error (B.32), we have

$$\begin{aligned}\epsilon_{\text{exc}} &\leq \epsilon_{\text{ave-stab}} + \epsilon_{\text{opt}} \\ &\leq \frac{2L^2 m(1+\beta n\alpha-\lambda)}{\beta^2 n\alpha(1-\lambda)}\frac{(1+\frac{\beta\alpha}{m})^T}{T} + \frac{Lmr}{T\gamma\alpha} + \frac{\beta L^2\alpha}{2\gamma}\left[\frac{1}{1-\lambda} + \frac{\overline{\tau}}{m}\right] + \frac{\beta L^2\alpha}{4\gamma m}.\end{aligned}$$

For the decreasing learning rate $\alpha_t = \frac{mc}{t+1}$, it follows from (B.27)

$$\begin{aligned}\mathbb{E}\|\overline{\mathbf{x}}_T - \overline{\mathbf{x}}'_T\| &\leq \frac{\sum_{t=1}^{T} \frac{mc}{t+1}\left[\frac{2L(1+\beta nmc-\lambda)}{\beta n(1-\lambda)}\right] t^{\beta c}}{\sum_{t=1}^{T}\frac{mc}{t+1}} \\ &\overset{(a)}{\leq} \left[\frac{4L(1+\beta nmc-\lambda)}{\beta n(1-\lambda)}\right]\frac{\sum_{t=1}^{T}(t+1)^{\beta c-1}}{\ln(T+1)} \\ &\overset{(b)}{\leq} \left[\frac{4L(1+\beta nmc-\lambda)}{\beta^2 nc(1-\lambda)}\right]\frac{(T+1)^{\beta c}}{\ln(T+1)},\end{aligned}$$

where (a) uses inequality (B.29). With $c < \frac{1}{\beta}$, (b) follows from

$$\sum_{t=1}^{T}(t+1)^{\beta c-1} \leq \sum_{t=1}^{T}\int_{t}^{t+1} x^{\beta c-1}dx \leq \int_{1}^{T+1} x^{\beta c-1}dx = \frac{1}{\beta c}(T+1)^{\beta c}.$$

Then the uniform stability of $\overline{\mathbf{x}}_T$ satisfies

$$\epsilon_{\text{ave-stab}} \leq \frac{4L^2}{\beta c}\left(\frac{1}{\beta n}+\frac{mc}{1-\lambda}\right)\frac{(T+1)^{\beta c}}{\ln(T+1)}.$$

Combined with the optimization error (B.33), we have

$$\begin{aligned}\epsilon_{\text{exc}} &\leq \epsilon_{\text{ave-stab}} + \epsilon_{\text{opt}} \\ &\leq \frac{4L^2}{\beta c}\left(\frac{1}{\beta n}+\frac{mc}{1-\lambda}\right)\frac{(T+1)^{\beta c}}{\ln(T+1)} + \left[2Lr + \beta mL^2c^2\left(\frac{C_\lambda}{\lambda^{\overline{\tau}}}+\frac{2\overline{\tau}}{m}\right)+\frac{\beta L^2c^2}{2}\right]\frac{1}{\gamma c\ln(T+1)}.\end{aligned}$$

■

References

Hardt, M.; Recht, B.; and Singer, Y. 2016. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, 1225–1234. PMLR.

Sun, T.; Li, D.; and Wang, B. 2021. Stability and generalization of decentralized stochastic gradient descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 9756–9764.

Lian, X.; Zhang, W.; Zhang, C.; and Liu, J. 2018. Asynchronous decentralized parallel stochastic gradient descent. In *International Conference on Machine Learning*, 3043–3052. PMLR.

Regatti, J.; Tendolkar, G.; Zhou, Y.; Gupta, A.; and Liang, Y. 2019. Distributed SGD generalizes well under asynchrony. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 863–870. IEEE.