Decentralized SGD and Average-direction SAM are Asymptotically Equivalent

Tongtian Zhu [1] **Fengxiang He** [2 3] **Kaixuan Chen** [1] **Mingli Song** [1] **Dacheng Tao** [4]

Abstract

Decentralized stochastic gradient descent (D-SGD) allows collaborative learning on massive devices simultaneously without the control of a central server. However, existing theories claim that decentralization invariably undermines generalization. In this paper, we challenge the conventional belief and present a completely new perspective for understanding decentralized learning. We prove that D-SGD implicitly minimizes the loss function of an average-direction Sharpness-aware minimization (SAM) algorithm under general non-convex non-β-smooth settings. This surprising asymptotic equivalence reveals an intrinsic regularization-optimization trade-off and three advantages of decentralization: (1) there exists a free uncertainty evaluation mechanism in D-SGD to improve posterior estimation; (2) D-SGD exhibits a gradient smoothing effect; and (3) the sharpness regularization effect of D-SGD does not decrease as total batch size increases, which justifies the potential generalization benefit of D-SGD over centralized SGD (C-SGD) in large-batch scenarios. Experiments support our theory and the code is available at D-SGD and SAM.

1. Introduction

Decentralization is a promising paradigm for harnessing the power of locally connected computing resources while preserving privacy (Warnat-Herresthal et al., 2021; Bhatia & Samet, 2022; Yuan et al., 2022). Decentralized stochastic gradient descent (D-SGD) is a popular decentralized algorithm which enables simultaneous model training on a massive number of workers without the need for a central server (Xiao & Boyd, 2004; Lopes & Sayed, 2008; Nedic & Ozdaglar, 2009; Lian et al., 2017; Koloskova et al., 2020). In D-SGD, each worker communicates only with its directly connected neighbors; see a detailed background in Appendix A.2. This decentralization avoids the requirements of a costly central server with heavy communication burdens and potential privacy risks. The existing literature demonstrates that the massive models on edge can converge to a unique steady consensus model even in the absence of a central server (Shi et al., 2015; Lu et al., 2011), with asymptotic linear speedup in convergence rate (Lian et al., 2017) similar to distributed centralized SGD (C-SGD) (Dean et al., 2012; Li et al., 2014). Therefore, D-SGD offers a promising distributed learning solution with significant advantages in communication efficiency (Ying et al., 2021b), privacy (Nedic, 2020) and scalability (Lian et al., 2017).

Despite these merits, it is regrettable that the existing theories claim decentralization to invariably undermines generalization (Sun et al., 2021; Zhu et al., 2022; Deng et al., 2023), which contradicts the following unique phenomena in decentralized deep learning 🤔:

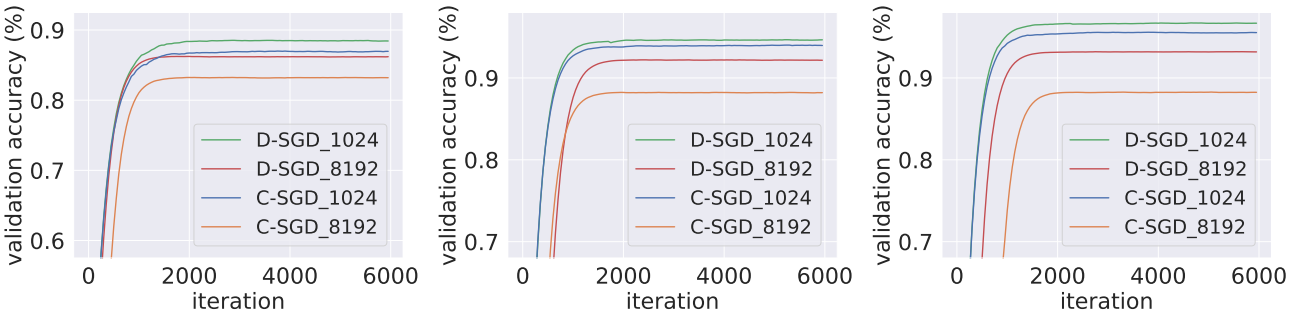
- D-SGD can outperform C-SGD in large-batch settings, achieving higher validation accuracy and smaller validation-training accuracy gap, despite both being fine-tuned (Zhang et al., 2021);
- A non-negligible consensus distance (see Equation (3)) at middle phases of decentralized training can improve generalization over centralized training (Kong et al., 2021).

These unexplained phenomena indicates the existence of a non-negligible **gap** between existing theories and deep learning experiments, which we attribute to the overlook of important characteristics of decentralized learning in existing literature. Accordingly, the central **motivation** of our study is to thoroughly examine the unique, underexamined characteristics of decentralized learning to bridge the gap.

Directly analyzing the dynamics of the diffusion-like decentralized learning systems, instead of relying on upper bounds, can be challenging. Instead, we aim to establish a relationship between D-SGD and other algorithms used for centralized training. In recent years, there has been a growing interest in techniques that aim to improve the generalization of deep learning models. One of the most popular techniques is sharpness-aware minimization (SAM) (Foret

[1]College of Computer Science and Technology, Zhejiang University [2]JD Explore Academy, JD.com, Inc. [3]Artificial Intelligence and its Applications Institute, School of Informatics, University of Edinburgh [4]The University of Sydney. Correspondence to: Fengxiang He <F.He@ed.ac.uk>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).



(a) AlexNet (b) ResNet-18 (c) DenseNet-121

Figure 1. The validation accuracy comparison of C-SGD and D-SGD (ring topology) on CIFAR-10. The number of workers is set as 16, with a local batch size of 64 and 512 per worker, resulting in total batch sizes of 1024 and 8192, respectively. Validation accuracy comparison of C-SGD and D-SGD with other topologies and on Tiny ImageNet are included in Figure B.1 and Figure B.2, respectively. The training accuracy is almost 100% everywhere. Exponential moving average is employed to smooth the validation accuracy curves. The training setting is included in Appendix B.

et al., 2021; Kwon et al., 2021; Zhuang et al., 2022; Du et al., 2022; Kim et al., 2022a), which is designed to explicitly minimize a sharpness-based perturbed loss; see the detailed background of SAM in Appendix A.3. Empirical studies have shown that SAM substantially improves the generalization of multiple architectures, including convolutional neural networks (Wu et al., 2020; Foret et al., 2021), vision transformers (Dosovitskiy et al., 2021) and large language models (Bahri et al., 2022). Average-direction SAM (Wen et al., 2023), a kind of SAM variants where sharpness is calculated as the (weighted) average within a small neighborhood around the current iterate, has been shown to generalize on par with vanilla SAM (Ujváry et al., 2022).

In this paper, we provide a completely new perspective for understanding decentralized learning by showing that

D-SGD and average-direction SAM are asymptotically equivalent.

Specifically, our contributions are summarized below.

- We prove that D-SGD asymptotically minimizes the loss function of an average-direction sharpness-aware minimization algorithm with zero additional computation (see Theorem 1), which directly connects decentralized learning and centralized learning. The asymptotic equivalence also implies a regularization-optimization trade-off in decentralized learning. Our theory is applicable to arbitrary communication topologies (see Definition A.1) and general **non-convex and non-β-smooth** (see Definition A.5) objectives (e.g., deep neural networks training).
- The equivalence further reveals the potential benefits of the decentralized learning paradigm, which challenges the previously held belief that centralized learning is optimal. We demonstrate three advantages of learning with decentralized models based on the equivalence: (1) there exists a surprising free uncertainty estimation mechanism in D-SGD, where the weight diversity matrix $\mathbf{\Xi}_{(t)}$ is learned to estimate Σ_q, the intractable covariance of the posterior;(2) D-SGD has a gradient smoothing effect, which improves training stability (see Corollary 2); and that (3) the sharpness regularizer of D-SGD does not decrease as the total batch size increases (see Theorem 3), which justifies the superior generalizability of D-SGD than C-SGD, especially in large-batch settings where gradient variance remains low. Our empirical results also fully support our theory (see Figure 1 and Figure 3).

To the best of our knowledge, our work is the first to establish the equivalence of D-SGD and average-direction SAM, which constitutes a direct connection between decentralized learning and centralized learning algorithms. This breakthrough makes it easier to analyze the diffusion-like decentralized systems, whose exact dynamics were considered challenging to understand. The theory further sheds light on the potential benefits of decentralized learning paradigm. While our theory primarily focuses on vanilla D-SGD, it can be directly extended to general decentralized gradient-based algorithms. We anticipate the insights derived from our work will help bridge the decentralized learning and SAM communities, and pave the way for the development of fast and generalizable decentralized learning algorithms.

2. Related work

Flatness and generalization. The flatness of the minimum has long been regarded as a proxy of generalization in the machine learning literature (Hochreiter & Schmidhuber, 1997; Izmailov et al., 2018; Jiang et al., 2020). Intuitively, the loss around a flat minimum varies slowly in a large neighborhood, while a sharp minimum increases rapidly in a small neighborhood (Hochreiter & Schmidhuber, 1997). Through the lens of the minimum description length theory

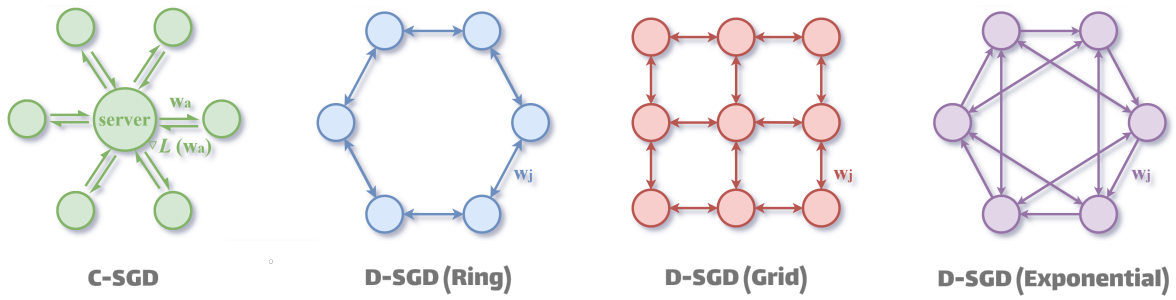


Figure 2. An illustration of centralized SGD and decentralized SGD. In C-SGD there is a central server aggregating global information while D-SGD relies only on peer-to-peer communication to diffuse information among workers.

(Rissanen, 1983), flat minimizers tend to generalize better than sharp minimizers, since they are specified with lower precision (Keskar et al., 2017). From a Bayesian perspective, sharp minimizers have posterior distributions that are highly concentrated around them, indicating that they are more specialized on the training set and thus are less robust to data perturbations than flat minimizers (MacKay, 1992; Chaudhari et al., 2019).

Generalization of D-SGD. Recently works by Sun et al. (2021), Zhu et al. (2022) and Deng et al. (2023) prove that decentralization introduces an additional positive term into the generalization error bounds, which suggests that decentralization may hurt generalization. However, these conservative upper bounds do not account for the unique phenomena in decentralized learning, such as the superior generalization performance of D-SGD in large batch settings. Gurbuzbalaban et al. (2022) offers an alternative perspective, showing that D-SGD with large, sparse topology has a heavier tail in parameter distribution than C-SGD in some cases, indicating better generalizability potential. Compared with Gurbuzbalaban et al. (2022), our theory is generally applicable to arbitrary communication topologies and learning rates. Another work by Zhang et al. (2021) demonstrates that decentralization introduces a landscape-dependent noise, which may improve the convergence of D-SGD. However, the impact of noise on the local geometry of the D-SGD trajectory and its effect on generalization remains unexplored. In contrast, we theoretically justify the flatness-seeking behavior of Hessian-dependent noise in D-SGD and then establish the asymptotic equivalence between D-SGD and SAM. Based on the equivalence, we prove that D-SGD has superior potential in generalizability compared with C-SGD, especially in large-batch settings.

3. Preliminaries

Suppose that $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ and $\mathcal{Y} \subseteq \mathbb{R}$ are the input and output spaces, respectively. We denote the training set as $\mu = \{z_1, \ldots, z_N\}$, where $z_\zeta = (x_\zeta, y_\zeta)\,, \zeta = 1, \ldots, N$ are sampled independent and identically distributed (i.i.d.) from an unknown data distribution $\mathcal{D}$ defined on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. The goal of supervised learning is to learn a predictor (or hypothesis) $g(\mathbf{w}; \cdot)$, parameterized by $\mathbf{w} \in \mathbb{R}^d$ of an arbitrary finite dimension d, to approximate the mapping between the input variable $x \in \mathcal{X}$ and the output variable $y \in \mathcal{Y}$, based on the training set μ. We define the function $c : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}^+$ as a function that evaluates the prediction performance of hypothesis g. The loss of a hypothesis g with respect to (w.r.t.) the example $z_\zeta = (x_\zeta, y_\zeta)$ is denoted by $\boldsymbol{L}(\mathbf{w}; z_\zeta) = c(g(\mathbf{w}; x_\zeta), y_\zeta)$, which measures the effectiveness of the learned model.The empirical and population risks of $\mathbf{w}$ are then defined as follows:

$$\boldsymbol{L}^{\mu}_{\mathbf{w}} = \frac{1}{N}\sum_{\zeta=1}^{N} \boldsymbol{L}(\mathbf{w}; z_\zeta), \quad \boldsymbol{L}_{\mathbf{w}} = \mathbb{E}_{z\sim D}[\boldsymbol{L}(\mathbf{w}; z)].$$

Distributed learning. Distributed learning jointly trains a model $\mathbf{w}$ using multiple workers (Shamir & Srebro, 2014). In this framework, the j-th worker (j=1, $\ldots, m$) can access training examples $\mu_j = \{z_{j,1}, \ldots, z_{j,|\mu_j|}\}$, independent and identically distributed (i.i.d.) drawn from the data distribution $\mathcal{D}$. In this setup, the global empirical risk of $\mathbf{w}$ becomes

$$\boldsymbol{L}^{\mu}_{\mathbf{w}} = \frac{1}{m}\sum_{j=1}^{m} \boldsymbol{L}^{\mu_j}_{\mathbf{w}},$$

where $\boldsymbol{L}^{\mu_j}_{\mathbf{w}} = \frac{1}{|\mu_j|}\sum_{\zeta=1}^{|\mu_j|} \boldsymbol{L}(\mathbf{w}; z_{j,\zeta})$ denotes the local empirical risk on the j-th worker and $z_{j,\zeta} \in \mu_j$, where $\zeta = 1, \ldots, |\mu_j|$, represent the local training data.

Distributed centralized stochastic gradient descent (C-SGD).[1] In C-SGD (Dean et al., 2012; Li et al., 2014), the

[1]"Centralized" refers to the fact that in C-SGD, there is a central server receiving local weights or gradients information (see Figure 2). C-SGD defined above is identical to the FedAvg algorithm (McMahan et al., 2017) under the condition that the local step is set as 1 and all local workers are selected by the server in each round (see Appendix A.1). To avoid misunderstandings, we include the term "distributed" in C-SGD to differentiate it from traditional single-worker SGD (Cauchy et al., 1847; Robbins, 1951).

de facto distributed training algorithm, there is only one centralized model $\mathbf{w}_a(t)$. C-SGD updates the model by

$$\mathbf{w}_a(t+1) = \mathbf{w}_a(t) - \eta \cdot \frac{1}{m} \sum_{j=1}^{m} \overbrace{\nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_a(t))}^{\text{Local gradient computation}}, \quad (1)$$

where η denotes the learning rate (step size), $\mu_j(t) = \{z_{j,1}, \ldots, z_{j,|\mu_j(t)|}\}$ denotes the local training batch independent and identically distributed (i.i.d.) drawn from the data distribution $\mathcal{D}$ at the t-th iteration, and $\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu_j(t)} = \nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}) = \frac{1}{|\mu_j(t)|} \sum_{\zeta(t)=1}^{|\mu_j(t)|} \nabla \boldsymbol{L}(\mathbf{w}; z_{j,\zeta(t)})$ stacks for the local mini-batch gradient of $\boldsymbol{L}$ w.r.t. the first argument $\mathbf{w}$. The total batch size of C-SGD at the t-th iteration is $|\mu(t)| = \sum_{j=1}^{m} |\mu_j(t)|$. Please refer to Appendix A.1 for more details of (distributed) centralized learning. In the next section, we will show that C-SGD equals the single-worker SGD with a larger batch size.

Decentralized stochastic gradient descent (D-SGD). In model decentralization scenarios, only peer-to-peer communication is allowed. The goal of D-SGD in the setup is to learn a consensus model $\mathbf{w}_a(t) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{w}_j(t)$ on m workers through gossip communication, where $\mathbf{w}_j(t)$ stands for the d-dimensional local model on the j-th worker. We denote $\mathbf{P} = [\mathbf{P}_{j,k}] \in \mathbb{R}^{m \times m}$ as a doubly stochastic gossip matrix (see Definition A.1) that characterizes the underlying topology $\mathcal{G}$. The vanilla Adapt-While-Communicate (AWC) version of the mini-batch D-SGD (Nedic & Ozdaglar, 2009; Lian et al., 2017) updates the model on the j-th worker by

$$\mathbf{w}_j(t+1) = \overbrace{\sum_{k=1}^{m} \mathbf{P}_{j,k} \mathbf{w}_k(t)}^{\text{Communication}} - \eta \cdot \overbrace{\nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_j(t))}^{\text{Local gradient computation}}. \quad (2)$$

For a more detailed background of decentralized learning, please kindly refer to Appendix A.2.

4. Theoretical results

In this section, we establish the equivalence between D-SGD and average-direction SAM under general non-convex and non-β-smooth objectives $\boldsymbol{L}$. We also provide a proof sketch to offer a more intuitive understanding. The equivalence further showcases the potential superiority of learning with decentralized models. Specifically, we prove that the additional noise introduced by decentralization leads to a gradient smoothing effect, which could stabilize optimization. Additionally, we show that the sharpness regularizer in D-SGD does not decrease as the total batch size increases, which guarantees generalizability in large-batch settings.

4.1. Equivalences of Decentralized SGD and average-direction SAM

In this subsection, we prove that D-SGD implicitly performs average-direction sharpness-aware minimization (SAM), followed by detailed implications.

Theorem 1 (D-SGD as SAM)**.** *Given the loss $\boldsymbol{L}$ is continuous and has fourth-order partial derivatives. The mean iterate[2] of the global averaged model of D-SGD[3], defined by $\mathbf{w}_a(t) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{w}_j(t)$, is provided as follows:*

$$\begin{aligned}
&\mathbb{E}_{\mu(t)}[\mathbf{w}_a(t+1)] \\
&= \mathbf{w}_a(t) - \eta \underbrace{\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]}_{\textit{asymptotic descent direction}} \\
&\quad \underbrace{+ \mathcal{O}(\eta\, \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))} \|\epsilon\|_2^3 + \frac{\eta}{m} \sum_{j=1}^{m} \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3)}_{\textit{higher-order residual terms}},
\end{aligned}$$

where $\boldsymbol{\Xi}(t) = \frac{1}{m} \sum_{j=1}^{m} (\mathbf{w}_j(t) - \mathbf{w}_a(t))(\mathbf{w}_j(t) - \mathbf{w}_a(t))^\top \in \mathbb{R}^{d \times d}$ denotes the weight diversity matrix.

The proof is deferred to Appendix C.2.

Asymptotic equivalence. According to Proposition C.3, the highlighted term $\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]$ is of the order $\mathcal{O}(\frac{1}{m} \sum_{j=1}^{m} \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^2)$ while the residual terms are of the higher-order $\mathcal{O}(\frac{1}{m} \sum_{j=1}^{m} \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3)$. Therefore, $\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]$ gradually dominates the optimization direction as the local models are near consensus (i.e., $\mathbf{w}_j(t) \to \mathbf{w}_a(t), \forall j$) and the descent direction of D-SGD asymptotically approaches $\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}]$. See the introduction of the asymptotic equivalence in Definition C.1.

Sharpness regularization. According to Theorem 1, D-SGD asymptotically optimizes $\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon}]$, an averaged perturbed loss in a "basin" around $\mathbf{w}$, rather than the original point-loss. To further clarify, we can split "true objective" of D-SGD near consensus into the original loss plus an average-direction sharpness:

$$\mathbb{E}_{\mu(t)}[\boldsymbol{L}_{\mathbf{w}}^{\text{D-SGD}}] \approx \underbrace{\boldsymbol{L}_{\mathbf{w}}}_{\textit{original loss}} + \underbrace{\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon} - \boldsymbol{L}_{\mathbf{w}}]}_{\textit{sharpness-aware regularizer}}.$$

The second term $\mathbb{E}_{\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon} - \boldsymbol{L}_{\mathbf{w}}]$ measures the weighted average sharpness at $\mathbf{w}$, which is a special form

[2]The expectation of the super batch $\mu(t)$ is computed by taking the expectations over all local mini-batches $\mu_j(t)$ for all $j=1, \ldots, m$, which eliminates the randomness of all training data at t-th iteration, represented by $z_{j,\zeta(t)}$ for all $\zeta(t)=1, \ldots, |\mu_j(t)|$ and $j=1, \ldots, m$.

[3]In D-SGD (see Equation (2)), the "virtual" global averaged model $\mathbf{w}_a(t) = \frac{1}{m} \sum_{j=1}^{m} \mathbf{w}_j(t)$ is primarily employed for theoretical analysis, since there is no central server to aggregate the information from local workers. However, analyzing $\mathbf{w}_a(t)$ remains practical as it characterizes the overall performance of D-SGD.

of the average-direction sharpness (Wen et al., 2023). Theorem 1 proves that D-SGD minimizes the loss function of an average-direction SAM asymptotically[4], which provides a direct connection between decentralized learning and centralized learning. As Theorem 1 only assumes $\boldsymbol{L}$ to be continuous and fourth-order differentiable, the result is generally applicable to **non-convex non-β-smooth** problems, including deep neural networks training.

We note that the sharpness regularizer in D-SGD is directly controlled by $\boldsymbol{\Xi}_{(t)}$, whose magnitude can be measured by the *consensus distance*, a key component characterizing the overall convergence of D-SGD (Kong et al., 2021),

$$\mathrm{Tr}\left(\boldsymbol{\Xi}_{(t)}\right) = \frac{1}{m}\sum_{j=1}^{m}\left(\mathbf{w}_{j(t)}-\mathbf{w}_{a(t)}\right)^{\top}\left(\mathbf{w}_{j(t)}-\mathbf{w}_{a(t)}\right). \quad (3)$$

Theorem 1 provides a theoretical explanation for the phenomena observed in (Kong et al., 2021): (1) Controlling consensus distance below a threshold in the initial training phases makes the SAM-type term quickly dominates the residual terms, thus ensuring good optimization; (2) Sustaining a non-negligible consensus distance at the middle phases maintains the sharpness regularization effect and therefore improves generalization over centralized training.

The implicit regularization effect in D-SGD shares similar insights with interesting studies on local SGD and federated learning, revealing that global coherence is not always optimal and a certain degree of drift in clients could be benign (Gu et al., 2023; Chor et al., 2023). Specifically, Gu et al. (2023) proves that the dissimilarity between local models induces a gradient noise, which drives the iterate to drift faster to flatter minima. Despite the shared characteristics, the consensus distance in decentralized learning is notably unique. The magnitude of the consensus distance exhibits dynamic adjustments. Proposition C.2 shows that if the learning rate is smaller than a certain threshold, then the consensus distance gradually decreases during training, indicating that the "searching space" of ϵ is relatively large in the initial training phase and then gradually declines. In addition, as shown in Proposition C.1, a small spectral gap (see Definition A.2) of the underlying communication topology (see Figure 2) leads to larger consensus distance, the magnitude of perturbation radius. According to (Foret et al., 2021), a large perturbation radius ensures a lower generalization upper bound. However, the validation performance of D-SGD is not always satisfactory on large and sparse topologies with a small spectral gap (Kong et al., 2021), as there is regularization-optimization trade-off in D-SGD (please refer to the discussion in Section 6).

[4]The generalization benefit of optimizing average-direction sharpness is discussed in Theorem C.6. This work reveals the sharpness regularization effect of D-SGD in one-step. We leave the study of the multi-step dynamics of D-SGD to future work.

Variational interpretation of D-SGD. In the variational formulation (Zellner, 1988), $\min_{\mathbf{w}} \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon}]$ is referred to as the variational optimization (Rockafellar & Wets, 2009). Theorem 1 shows that D-SGD not only optimizes $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}$ with respect to $\mathbf{w}$, but also inherently estimates uncertainty for free: The weight diversity matrix $\boldsymbol{\Xi}_{(t)}$ (i.e., the empirical covariance matrix of $\mathbf{w}_{j(t)}$) implicitly estimate Σ_q, the intractable posterior covariance,

$$\boldsymbol{\Xi}_{(t)} = \frac{1}{m}\sum_{j=1}^{m}\left(\mathbf{w}_{j(t)}-\mathbf{w}_{a(t)}\right)\left(\mathbf{w}_{j(t)}-\mathbf{w}_{a(t)}\right)^{\top} \approx \Sigma_q.$$

Note that $\boldsymbol{\Xi}_{(t)}$ is learned along with the update of local models without any additional computational budget. The free uncertainty estimation mechanism indicates the uniqueness of the noise from decentralization, which depends both on the local loss landscape and the posterior distribution.

Comparison of D-SGD and vanilla SAM. The loss function of vanilla SAM (Foret et al., 2021) can be written in the following form:

$$\boldsymbol{L}^{\text{SAM}}(\mathbf{w},\Sigma)=\max_{\epsilon\in\mathbb{R}^d|\epsilon^T\Sigma^{-1}\epsilon\leq d}[\boldsymbol{L}(\mathbf{w}+\epsilon)-\boldsymbol{L}(\mathbf{w})]+\boldsymbol{L}(\mathbf{w}),$$

where the covariance matrix Σ is set as $\frac{\rho^2}{d}I$, with ρ being the perturbation radius and I representing the identity matrix. Interestingly, ρ in SAM plays a similar role as $\boldsymbol{\Xi}_{(t)}$ in D-SGD. However, in the SAM that D-SGD approximates, the covariance matrix $\boldsymbol{\Xi}_{(t)}$ is learned adaptive during training. Moreover, the iterate of D-SGD involves higher-order residuals, whereas vanilla SAM does not. The third difference is that vanilla SAM minimizes a worst-case sharpness $\max_{\epsilon^T\Sigma^{-1}\epsilon\leq d} L(\mathbf{w}+\epsilon)$ while D-SGD implicitly minimizes an average-direction sharpness (or a Bayes loss). However, the loss of vanilla SAM always upper bounds the Bayes loss (Möllenhoff & Khan, 2023), and they are close to each other in high dimensions where samples from $\mathcal{N}(\mathbf{w},\Sigma)$ concentrate around the ellipsoid $(\mathbf{w}-\epsilon)^{\top}\Sigma^{-1}(\mathbf{w}-\epsilon)=d$ (Vershynin, 2018). In addition, the sharpness regularization effect of D-SGD incurs **zero** additional computational overhead compared to SAM, which requires performing backpropagation twice at each iteration.

Comparison with related works. Initial efforts have viewed D-SGD as a centralized algorithm penalizing the weight norm $\|\mathbf{P}^{-\frac{1}{2}}\mathbf{W}\|^2_{\mathbf{I}-\mathbf{P}}$, a quantity similar to the consensus distance, where $\mathbf{W}=[\mathbf{w}_1,\cdots,\mathbf{w}_m]^T\in\mathbb{R}^{m\times d}$ collects m local models (Yuan et al., 2021; Gurbuzbalaban et al., 2022). However, little effort has been made so far to analyze the "interplay" between weight diversity measures, such as the consensus distance, and the local geometry of the D-SGD iterate. Our work fills this gap by showing the flatness-seeking behavior of the Hessian-consensus dependent noise in D-SGD and then exhibiting the asymptotic equivalence between D-SGD and SAM.

4.2. Proof sketch and idea

To impart a stronger intuition, we summarize the proof sketch of Theorem 1 and explain the motivation behind our proof idea. Full proof is deferred to Appendix C.

(1) Derive the iterate of the averaged model $\mathbf{w}_{a(t)}$.

Directly analyzing the dynamics of the diffusion-like decentralized systems where information is gradually spread across the network is non-trivial. Instead, we focus on $\mathbf{w}_{a(t)} = \frac{1}{m}\sum_{j=1}^{m}\mathbf{w}_{j(t)}$, the global averaged model of D-SGD, whose update can be written as follows,

$$\mathbf{w}_{a(t+1)} = \mathbf{w}_{a(t)} - \eta\Big[\nabla \boldsymbol{L}^{\mu(t)}_{\mathbf{w}_{a(t)}} + \underbrace{\frac{1}{m}\sum_{j=1}^{m}\big(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{j(t)}} - \nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}\big)}_{\textit{gradient diversity among local workers}}\Big], \quad (4)$$

as $\frac{1}{m}\sum_{j=1}^{m}\sum_{k=1}^{m}\mathbf{P}_{j,k}\mathbf{w}_{k(t)} = \frac{1}{m}\sum_{k=1}^{m}\mathbf{w}_{k(t)} = \mathbf{w}_{a(t)}$.

Equation (4) shows that decentralization introduces an additional noise, which characterizes the gradient diversity[5] between the global averaged model $\mathbf{w}_{a(t)}$ and the local models $\mathbf{w}_{j(t)}$ for $j=1,\ldots,m$, compared with its centralized counterpart. Therefore, we note that

> *analyzing the gradient diversity is the major challenge of decentralized (gradient-based) learning.*

One can deduce directly from Equation (4) that distributed centralized SGD, which has constant zero gradient diversity, is equivalent to standard single-worker mini-batch SGD with an equivalently large batch size.

Insight. We also note that the gradient diversity equals to zero on quadratic objective $\boldsymbol{L}$ (see Proposition C.4). Therefore, the quadratic approximation of loss functions $\boldsymbol{L}$ (Zhu et al., 2019b; Ibayashi & Imaizumi, 2021; Liu et al., 2021; 2022c) is insufficient to characterize how decentralization impacts the training dynamics of D-SGD, especially on neural network loss landscapes where quadratic approximation may not be accurate even around minima (Ma et al., 2022). To better understand the dynamics of D-SGD on complex landscapes, it is crucial to consider *higher-order geometric information* of objective $\boldsymbol{L}$. In the following, we approximate the gradient diversity using Taylor expansion, instead of analyzing it on non-convex non-β-smooth loss $\boldsymbol{L}$ directly, which is highly non-trivial.

(2) Perform Taylor expansion on the gradient diversity. Technically, we perform a second-order Taylor expansion on the gradient diversity around $\mathbf{w}_{a(t)}$:

$$\frac{1}{m}\sum_{j=1}^{m}\big(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{j(t)}} - \nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}\big) = \frac{1}{m}\sum_{j=1}^{m}\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}\cdot\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big) + \frac{1}{2m}\sum_{j=1}^{m}\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_{a(t)}} \otimes \big[\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)^{\top}\big],$$

plus residual terms $\mathcal{O}(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\|_2^3)$. Here $\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_{a(t)}} \triangleq \frac{1}{|\mu_j(t)|}\sum_{\zeta(t)=1}^{|\mu_j(t)|}\boldsymbol{H}(\mathbf{w}_{a(t)}; z_{j,\zeta(t)})$ denotes the empirical Hessian matrix evaluated at $\mathbf{w}_{a(t)}$ and $\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_{a(t)}} \triangleq \frac{1}{|\mu_j(t)|}\sum_{\zeta(t)=1}^{|\mu_j(t)|}\boldsymbol{T}(\mathbf{w}_{a(t)}; z_{j,\zeta(t)})$ stacks for the empirical third-order partial derivative tensor at $\mathbf{w}_{a(t)}$, where $\mu_j(t)$ and $z_{j,\zeta(t)}$ follows the notation in Equation (1).

As $\mathbf{w}_{a(t)}$ and local models $\mathbf{w}_{j(t)}$ $(j=1,\ldots,m)$ are only correlated with the super batch before the t-th iteration (see Equation (2)), taking expectation over $\mu_{(t)}$ provides

$$\mathbb{E}_{\mu(t)}\Big[\frac{1}{m}\sum_{j=1}^{m}\big(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{j(t)}} - \nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}\big)\Big] = \boldsymbol{H}_{\mathbf{w}_{a(t)}}\cdot\underbrace{\frac{1}{m}\sum_{j=1}^{m}\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)}_{=0} + \frac{1}{2}\boldsymbol{T}_{\mathbf{w}_{a(t)}} \otimes \Big[\frac{1}{m}\sum_{j=1}^{m}\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)^{\top}\Big],$$

plus residual terms $\mathcal{O}(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\|_2^3)$, where $\boldsymbol{H}_{\mathbf{w}_{a(t)}} = \mathbb{E}_{\mu_j(t)}[\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}]$ and $\boldsymbol{T}_{\mathbf{w}_{a(t)}} = \mathbb{E}_{\mu_j(t)}[\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}]$.

The i-th entry of the expected gradient diversity becomes:

$$\mathbb{E}_{\mu(t)}\Big[\frac{1}{m}\sum_{j=1}^{m}\big(\partial_i \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{j(t)}} - \partial_i \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_{a(t)}}\big)\Big] = \underbrace{\frac{1}{2}\sum_{l,s}\partial^3_{ils}\boldsymbol{L}_{\mathbf{w}_{a(t)}}\frac{1}{m}\sum_{j=1}^{m}\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)_l\big(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\big)_s}_{=\partial_i\sum_{ls}\partial^2_{ls}\boldsymbol{L}_{\mathbf{w}_{a(t)}}[\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)})_l(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)})_s]} + \mathcal{O}\Big(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}\|_2^3\Big), \quad (5)$$

where $(\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)})_l$ denotes the l-th entry of the vector $\mathbf{w}_{j(t)} - \mathbf{w}_{a(t)}$. The equality in the brace is due to Clairaut's theorem (Rudin et al., 1976). The right hand side (RHS) of this equality is the i-th partial derivative of

$$\begin{aligned}&\mathrm{Tr}(\nabla^2 \boldsymbol{L}_{\mathbf{w}_{a(t)}}\boldsymbol{\Xi}_{(t)})\\ &= \mathrm{Tr}(\nabla^2 \boldsymbol{L}_{\mathbf{w}_{a(t)}}\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}_{(t)})}[\epsilon\epsilon^T])\\ &= \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}_{(t)})}[\epsilon^T\nabla^2 \boldsymbol{L}_{\mathbf{w}_{a(t)}}\epsilon]\end{aligned}$$

[5] We note that the concept of gradient diversity is distinct from that in (Yin et al., 2018), as it quantifies the variation of the gradients of one single model on different data points. The gradient diversity in our paper shares similarities with the gradient bias of local workers in federated learning (FL) literature (Wang et al., 2020; Reddi et al., 2021; Wang et al., 2022).

$$= \underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[2(\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}-\boldsymbol{L}_{\mathbf{w}_a(t)})}_{\text{average-direction sharpness at } \mathbf{w}_a(t)}+\mathcal{O}(\|\epsilon\|_2^3)], \quad (6)$$

where $\boldsymbol{\Xi}(t)=\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))(\mathbf{w}_j(t)-\mathbf{w}_a(t))^T$.

Proof complete by combining Equation (4) and Equation (6). □

The proof sketch outlines the high-level intuition of the flatness-seeking behavior of D-SGD.

***High-level intuition**: Model decentralization introduces gradient diversity among local models (see Equation (4)), which induces a unique Hessian-consensus dependent noise. This noise directs the optimization trajectory of D-SGD towards regions with lower average-direction sharpness $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon}-\boldsymbol{L}_{\mathbf{w}}]$.*

4.3. Gradient smoothing effect of decentralization

Previous literature has shown the gradient stabilization effect of isotropic Gaussian noise injection (Bisla et al., 2022; Liu et al., 2022b). According to Theorem 1, decentralization can be interpreted as the injection of Gaussian noise into gradient. There arises a natural question that whether or not the noise introduced from decentralization, which is not necessarily isotropic, would smooth the gradient. We employ the following Corollary to answer this question.

Corollary 2 (Gradient smoothing effect of D-SGD)**.** *Given that vanilla loss function $\boldsymbol{L}_{\mathbf{w}}$ is α-Lipschitz continuous and the gradient $\nabla\boldsymbol{L}_{\mathbf{w}}$ is β-Lipschitz continuous. We conclude that the gradient $\nabla\boldsymbol{L}_{\mathbf{w}+\epsilon}$ where $\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))$ is $\min\left\{\frac{\sqrt{2}\alpha}{\sigma_{\min}},\beta\right\}$-Lipschitz continuous, where $\sigma_{\min}$ denotes the smallest eigenvalue of $\boldsymbol{\Xi}(t)$.*

Corollary 2 implies that if the lower bound of noise magnitude satisfies $\sigma_{\min}\geq\frac{\sqrt{2}\alpha}{\beta}$, then the noise ϵ can make the Lipschitz constant of gradients smaller, therefore leading to gradient smoothing. The gradient smoothing effect exhibited by D-SGD aligns with two empirical observations in large-batch settings: (1) the training curves of D-SGD are notably more stable than those of C-SGD, and (2) D-SGD can converge with larger learning rates, as reported in (Zhang et al., 2021). The proof is deferred to Appendix C.3. Further research directions include dynamical stability analysis (Kim et al., 2023; Wu & Su, 2023) of D-SGD.

4.4. Generalization benefit of D-SGD: the role of total batch size

In practice, data and model decentralization ordinarily imply large total batch sizes, as a massive number of workers are involved in the system in many practical scenarios. Large-batch training can enhance the utilization of super computing facilities and further speed up the entire training process. Thus, studying the large-batch setting is crucial for fully understanding the application of D-SGD.

Despite the importance, theoretical understanding of the generalization of large-batch[6] decentralized training remains an open problem. This subsection examines the implicit regularization of D-SGD with respect to (w.r.t.) total batch size $|\mu|$, and compares it to C-SGD. To investigate the impact of $|\mu|$, we analyze the effect of the gradient variance, in addition to the gradient expectation studied in Subsection 4.1.

Theorem 3. *Let $B=|\mu|$ denote the total batch size. With a probability greater than $1-\mathcal{O}(\frac{B}{(N-B)\eta^2})$, D-SGD implicit minimizes the following objective function:*

$$\boldsymbol{L}_{\mathbf{w}}^{\text{D-SGD}}=\boldsymbol{L}_{\mathbf{w}}^{\mu}+\underbrace{\mathrm{Tr}(\boldsymbol{H}_{\mathbf{w}}^{\mu}\boldsymbol{\Xi}(t))+\frac{\eta}{4}\mathrm{Tr}((\boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t))}_{\text{batch size independent sharpness regularizer}}$$

$$+\underbrace{\kappa\cdot\frac{1}{N}\sum_{j=1}^{N}\left[\|\nabla\boldsymbol{L}_{\mathbf{w}}^{j}-\nabla\boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2+\mathrm{Tr}((\boldsymbol{H}_{\mathbf{w}}^{j}-\boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t))\right]}_{\text{batch size dependent variance regularizer}}$$

$$+\frac{\eta}{4}\|\nabla\boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2+\mathcal{R}^A+\mathcal{O}(\eta^2),$$

where $\kappa=\frac{\eta}{B}\cdot\frac{N-B}{(N-1)}$, and $\mathcal{R}^A$ absorbs all higher-order residuals. The empirical loss and the gradient on the super-batch μ, denoted by $\boldsymbol{L}_{\mathbf{w}}^{\mu}$ and $\nabla\boldsymbol{L}_{\mathbf{w}}^{\mu}$, respectively, are averaged over the one-sample gradients $\nabla\boldsymbol{L}_{\mathbf{w}}^{j}$. Similarly, the empirical Hessian $\boldsymbol{H}_{\mathbf{w}}^{\mu}$ is an average of $\boldsymbol{H}_{\mathbf{w}}^{j}=\boldsymbol{H}(\mathbf{w};z_j)$.

A corresponding implicit regularization of C-SGD (and SGD) is established in Lemma C.8, which demonstrates that C-SGD has an implicit gradient variance reduction mechanism to improve generalization (see Figure C.1). However, as the total batch size B approaches the total training sample size N, the regularization term diminishes rapidly, even in the case when the learning rate scales with the total batch size, since the ratio $\kappa=\frac{N-B}{N-1}$ converges to 0 gradually.

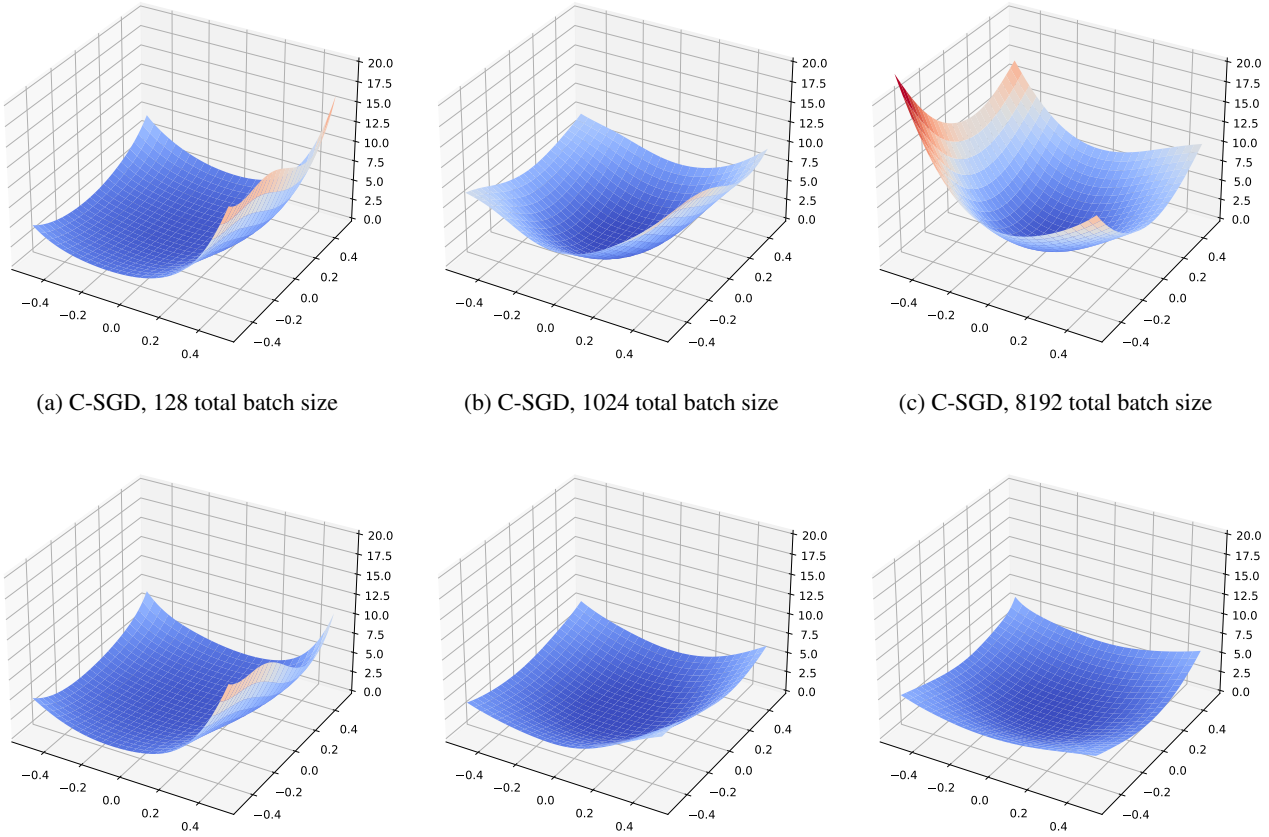
On the contrary, Theorem 3 proves that the sharpness regularization terms in D-SGD do not decrease as the total batch size increases, unlike in C-SGD, which theoretically justifies the potential superior generalizability of D-SGD in large-batch settings. The underlying intuition is that decentralization introduces additional noise, which compensates for the noise required for D-SGD to generalize well in large-batch scenarios. The proof is included in Appendix C.5.

5. Empirical results

This section empirically validates our theory [7]. We introduce the experimental setup and then study how decentralization

[6] Please refer to Appendix A.4 for the detailed discussion on the generalization of large-batch training.

[7] Code is available at D-SGD and SAM.



(a) C-SGD, 128 total batch size (b) C-SGD, 1024 total batch size (c) C-SGD, 8192 total batch size

(d) D-SGD, 128 total batch size (e) D-SGD, 1024 total batch size (f) D-SGD, 8192 total batch size

Figure 3. Minima 3D visualization of ResNet-18 trained on CIFAR-10 using C-SGD and D-SGD (ring topology).

impacts minima flatness via local landscape visualization.

Dataset and architecture. Decentralized learning is simulated in a dataset-centric setup by uniformly partitioning data among multiple workers (GPUs) to accelerate the training process. Vanilla D-SGD with various commonly used topologies[8] (see Figure 2) and C-SGD are employed to train image classifiers on CIFAR-10 (Krizhevsky et al., 2009) and Tiny ImageNet (Le & Yang, 2015) with AlexNet (Krizhevsky et al., 2017), ResNet-18 (He et al., 2016b) and DenseNet-121[9] (Huang et al., 2017). Detailed implementation setting is inclued in Appendix B.

As demonstrated in Figure 1, Figure B.1 and Figure B.2, D-SGD consistently outperforms C-SGD in terms of generalizability in large-batch settings[10]. We also note that the gap in generalizability between D-SGD and C-SGD in a large-batch scenario is larger on the CIFAR-10 dataset, which we attribute to the smaller κ value (see Theorem 3). To further support our claim that D-SGD favors flatter minima than C-SGD in large-batch scenarios, we plotted the minima learned by both algorithms using the local loss landscape 3D visualization tool in Li et al. (2018). The resulting plots are shown in Figure 3, along with additional plots in Appendix B. These plots demonstrate that D-SGD consistently learns flatter minima than C-SGD in large-batch settings, and this difference in flatness becomes larger as the total batch size increases. These observations are consistent with the claims made in Theorem 1 and Theorem 3 that D-SGD favors flatter minima than its centralized counterpart, especially in the large-batch scenarios. Future work includes visualizing the whole trajectories of D-SGD.

[8]On CIFAR-10, we use deterministic topology. On Tiny ImageNet, we use deterministic topology with random neighbor shuffling, which can increase the effective spectral gap of the underlying communication matrix (Zhang et al., 2020). We adjust the effective spectral gap by introducing random shuffle to accommodate the "optimal temperature" of models on different datasets.

[9]In our experiments, the ImageNet pre-trained models are used as initializations to achieve better final validation performance. The conclusions still stand for training from scratch.

[10]This is due to the fact that the training accuracy of D-SGD is almost surely 100% in all settings, making validation accuracy a reliable measure of generalizability.

6. Discussion and Broader Impact

Question 1: How to scale learning rate w.r.t. batch size and spectral gap in decentralized deep learning?

As proved in Theorem 1, the training dynamics of D-SGD and C-SGD are completely different. There also exists potential landscape smoothing and generalization promoting effect of D-SGD (Theorem C.6 and Theorem 3). Consequently, we conjecture that D-SGD could be more "tolerable" to hyperparameters such as learning rate than C-SGD. However, the existing tricks for hyperparameter tuning are tailored for C-SGD (e.g., linear scaling rule). A natural question is that can we design a general scaling strategy of learning rate as a function of batch size and the spectral gap of the underlying communication topology, in order to main generalizability of D-SGD in large-batch settings?

Question 2: Would continuously reducing spectral gap improve the validation performance of D-SGD?

The answer is negative, as D-SGD inherently exhibits a regularization-optimization trade-off. On the one hand, a sparse topology[11] with a too-small spectral gap leads to large consensus distance (see proof in Proposition C.1), which also slows down the converges of the higher-order residual terms, and consequently, hampers the optimization of the original objective. On the other hand, although a small spectral gap increases the sharpness regularization effect, it also makes the optimization of $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]$ difficult. In SAM, a too large perturbation radius (or neighborhood size) ρ incurs divergence (Andriushchenko & Flammarion, 2022; Mueller & Hein, 2022). Similarly, a topology with a small spectral gap leads to a large $\mathrm{Tr}\left(\boldsymbol{\Xi}_{(t)}\right)$, the magnitude of the variance of ϵ. A large $\mathrm{Tr}\left(\boldsymbol{\Xi}_{(t)}\right)$ results in an increased search space for the optimization problem $\min_{\mathbf{w}} \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon} - \boldsymbol{L}_{\mathbf{w}}]$, making it more difficult to solve. These intuitions align with the observation that the D-SGD converges slowly on sparse topologies (Lian et al., 2017; Koloskova et al., 2020). Based on the analysis, we conjecture that there exists "sweet spots", which balance the sharpness regularization introduced by proper decentralization and the optimization issues originated from sparse topologies. Finding these "sweet spots" then becomes a promising direction in a decentralized deep learning.

Future work 1: Fine-grained convergence and generalization analyses of decentralized learning algorithms.

We have demonstrated several potential benefits of D-SGD based on the established equivalence. Another direct extension of our work is to utilize the connection between decentralized learning and centralized learning to improve the existing convergence and generalization bounds of decentralized learning algorithms.

Future work 2: Provide adversarial robustness guarantees for decentralized learning algorithms.

Cao et al. (2023) find that decentralized stochastic gradient algorithms are more adversarial robust than their centralized counterpart in certain scenarios. They boldly conjecture that the superiority could be attributed to their inclination towards flatter minimizers. The question is can we provide rigorous adversarial robustness guarantees of decentralized learning algorithms, and develop more robust algorithms?

Future work 3: Bridge decentralized learning and SAM.

An interesting question arising from the asymptotic equivalence is that does D-SGD share the properties of SAM, beyond generalizablity, including better interpretability (Andriushchenko et al., 2023) and transferability (Chen et al., 2022)? Furthermore, it is worth exploring whether the insights gained from SAM (Andriushchenko & Flammarion, 2022; Möllenhoff & Khan, 2023; Wen et al., 2023) can be utilized to design more effective decentralized algorithms.

7. Conclusion

This paper challenges the conventional belief that centralized learning is optimal and establishes the surprising asymptotic equivalence of decentralized SGD and average-direction SAM. This asymptotic equivalence further demonstrates a regularization-optimization trade-off and three advantages of D-SGD: (1) there exists a surprising free uncertainty estimation mechanism in D-SGD to estimate the intractable posterior covariance;(2) D-SGD has a gradient smoothing effect; and (3) the sharpness regularization effect of D-SGD does not decrease as total batch size increases, which justifies the superior generalizablity of D-SGD over centralized SGD (C-SGD) in large-batch settings. Although our theory focuses primarily on the vanilla decentralized SGD, we believe our theoretical insights are applicable to a broad range of decentralized gradient-based algorithms.

Acknowledgement

This work is supported by National Natural Science Foundation of China (U20B2066, 61976186), the Zhejiang Provincial Key Research and Development Program of China (2021C01164), and the Fundamental Research Funds for the Central Universities (2021FZZX001-23, 226-2023-00048). Prof Dacheng Tao is partially supported by Australian Research Council Project FL-170100117.

The authors appreciate Xinran Gu for the valuable discussions on Local SGD and the limiting dynamics of D-SGD.

[11] A sparse topology refers to the topology whose neighbor number is relatively smaller than the total number of workers. A sparse topology always has a small spectral gap (see Definition A.2)

References

Alquier, P. User-friendly introduction to pac-bayes bounds. *arXiv preprint arXiv:2110.11216*, 2021.

Andriushchenko, M. and Flammarion, N. Towards understanding sharpness-aware minimization. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 639–668. PMLR, 2022.

Andriushchenko, M., Bahri, D., Mobahi, H., and Flammarion, N. Sharpness-aware minimization leads to low-rank features. *arXiv preprint arXiv:2305.16292*, 2023.

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

Bahri, D., Mobahi, H., and Tay, Y. Sharpness-aware minimization improves language model generalization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7360–7371. Association for Computational Linguistics, 2022.

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

Bhatia, L. and Samet, S. Decentralized federated learning: A comprehensive survey and a new blockchain-based data evaluation scheme. In *2022 Fourth International Conference on Blockchain Computing and Applications (BCCA)*, pp. 289–296, 2022.

Bisla, D., Wang, J., and Choromanska, A. Low-pass filtering sgd for recovering flat optima in the deep learning optimization landscape. In *International Conference on Artificial Intelligence and Statistics*, pp. 8299–8339. PMLR, 2022.

Bornstein, M., Rabbani, T., Wang, E. Z., Bedi, A., and Huang, F. SWIFT: Rapid decentralized federated learning via wait-free model communication. In *The Eleventh International Conference on Learning Representations*, 2023.

Bottou, L., Curtis, F. E., and Nocedal, J. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Cao, Y., Rizk, E., Vlaski, S., and Sayed, A. H. Decentralized adversarial training over graphs. *arXiv preprint arXiv:2303.13326*, 2023.

Cauchy, A. et al. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538, 1847.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019.

Chebyshev, P. L. *Sur les valeurs limites des intégrales*. Imprimerie de Gauthier-Villars Paris, 1874.

Chen, K. and Huo, Q. Scalable training of deep learning machines by incremental block training with intra-block parallel optimization and blockwise model-update filtering. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5880–5884. IEEE Press, 2016.

Chen, X., Hsieh, C.-J., and Gong, B. When vision transformers outperform resnets without pre-training or strong data augmentations. In *International Conference on Learning Representations*, 2022.

Chor, R., Sefidgaran, M., and Zaidi, A. More communication does not result in smaller generalization error in federated learning. *arXiv preprint arXiv:2304.12216*, 2023.

Davis, C. The norm of the schur product operation. *Numerische Mathematik*, 4:343–344, 1962.

De Bruijn, N. G. *Asymptotic methods in analysis*, volume 4. Courier Corporation, 1981.

Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., Ranzato, M., Senior, A., Tucker, P., Yang, K., et al. Large scale distributed deep networks. *Advances in neural information processing systems*, 2012.

Deng, X., Sun, T., Li, S., and Li, D. Stability-based generalization analysis of the asynchronous decentralized sgd. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

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

Du, J., Yan, H., Feng, J., Zhou, J. T., Zhen, L., Goh, R. S. M., and Tan, V. Efficient sharpness-aware minimization for improved training of neural networks. In *International Conference on Learning Representations*, 2022.

Dziugaite, G. K. and Roy, D. M. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Proceedings of the 33rd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

Erdélyi, A. *Asymptotic expansions*. Number 3. Courier Corporation, 1956.

Farhadkhani, S., Guerraoui, R., Gupta, N., Hoang, L. N., Pinot, R., and Stephan, J. Robust collaborative learning with linear gradient overhead. In *International Conference on Machine Learning*, 2023.

Fawcett, L. and Green, A. C. Bayesian posterior predictive return levels for environmental extremes. *Stochastic Environmental Research and Risk Assessment*, 32:2233–2252, 2018.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

Geiping, J., Bauermeister, H., Dröge, H., and Moeller, M. Inverting gradients - how easy is it to break privacy in federated learning? In *Advances in Neural Information Processing Systems*, 2020.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

Gu, X., Lyu, K., Huang, L., and Arora, S. Why (and when) does local SGD generalize better than SGD? In *International Conference on Learning Representations*, 2023.

Gurbuzbalaban, M., Hu, Y., Simsekli, U., Yuan, K., and Zhu, L. Heavy-tail phenomenon in decentralized sgd. *arXiv preprint arXiv:2205.06689*, 2022.

He, F., Liu, T., and Tao, D. Control batch size and learning rate to generalize well: Theoretical and empirical evidence. In *Advances in Neural Information Processing Systems*, pp. 1143–1152, 2019.

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

He, K., Zhang, X., Ren, S., and Sun, J. Identity mappings in deep residual networks. In *European conference on computer vision*. Springer, 2016b.

Hochreiter, S. and Schmidhuber, J. Flat minima. *Neural computation*, 9(1):1–42, 1997.

Hoffer, E., Hubara, I., and Soudry, D. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. *Advances in neural information processing systems*, 2017.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Ibayashi, H. and Imaizumi, M. Exponential escape efficiency of sgd from sharp minima in non-stationary regime. *arXiv preprint arXiv:2111.04004*, 2021.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 2015.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Jiang, Y., Neyshabur*, B., Mobahi, H., Krishnan, D., and Bengio, S. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2020.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

Kim, H., Park, J., Choi, Y., and Lee, J. Stability analysis of sharpness-aware minimization. *arXiv preprint arXiv:2301.06308*, 2023.

Kim, M., Li, D., Hu, S. X., and Hospedales, T. Fisher sam: Information geometry and sharpness aware minimisation. In *International Conference on Machine Learning*, pp. 11148–11161. PMLR, 2022a.

Kim, M., Li, D., Hu, S. X., and Hospedales, T. Fisher SAM: Information geometry and sharpness aware minimisation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 11148–11161. PMLR, 2022b.

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., and Stich, S. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, 2020.

Kong, L., Lin, T., Koloskova, A., Jaggi, M., and Stich, S. Consensus control for decentralized deep learning. In *International Conference on Machine Learning*. PMLR, 2021.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images (tech. rep.). *University of Toronto*, 2009.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Kwon, J., Kim, J., Park, H., and Choi, I. K. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International Conference on Machine Learning*, pp. 5905–5914. PMLR, 2021.

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

Le Bars, B., Bellet, A., Tommasi, M., Lavoie, E., and Kermarrec, A.-M. Refined convergence and topology learning for decentralized sgd with heterogeneous data. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206, pp. 1672–1702. PMLR, 25–27 Apr 2023.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 2018.

Li, M., Andersen, D. G., Smola, A. J., and Yu, K. Communication efficient distributed machine learning with the parameter server. *Advances in Neural Information Processing Systems*, 2014.

Li, Z., Malladi, S., and Arora, S. On the validity of modeling sgd with stochastic differential equations (sdes). *Advances in Neural Information Processing Systems*, 2021.

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

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

Liu, K., Ziyin, L., and Ueda, M. Noise and fluctuation of finite learning rate stochastic gradient descent. In *International Conference on Machine Learning*. PMLR, 2021.

Liu, Y., Mai, S., Chen, X., Hsieh, C.-J., and You, Y. Towards efficient and scalable sharpness-aware minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12360–12370, 2022a.

Liu, Y., Mai, S., Cheng, M., Chen, X., Hsieh, C.-J., and You, Y. Random sharpness-aware minimization. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022b.

Liu, Z., Kangqiao, L., Takashi, M., and Masahito, U. Strength of minibatch noise in SGD. In *International Conference on Learning Representations*, 2022c.

Lopes, C. G. and Sayed, A. H. Diffusion least-mean squares over adaptive networks: Formulation and performance analysis. *IEEE Transactions on Signal Processing*, 2008.

Lu, J., Tang, C. Y., Regier, P. R., and Bow, T. D. Gossip algorithms for convex consensus optimization over networks. *IEEE Transactions on Automatic Control*, 2011.

Lu, S. and Wu, C. W. Decentralized stochastic non-convex optimization over weakly connected time-varying digraphs. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Ma, C., Kunin, D., Wu, L., and Ying, L. Beyond the quadratic approximation: The multiscale structure of neural network loss landscapes. *Journal of Machine Learning*, 1(3):247–267, 2022.

MacKay, D. J. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.

Marshall, A. W. and Olkin, I. Multivariate chebyshev inequalities. *The Annals of Mathematical Statistics*, pp. 1001–1014, 1960.

Marshall, A. W., Olkin, I., and Arnold, B. C. *Inequalities: theory of majorization and its applications*. Springer, 1979.

McAllester, D. A. Pac-bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational learning theory*, pp. 164–170, 1999.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pp. 1273–1282. PMLR, 2017.

Merikoski, J. K. On the trace and the sum of elements of a matrix. *Linear algebra and its applications*, 60:177–185, 1984.

Mi, P., Shen, L., Ren, T., Zhou, Y., Sun, X., Ji, R., and Tao, D. Make sharpness-aware minimization stronger: A sparsified perturbation approach. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022.

Mueller, M. and Hein, M. Perturbing batchnorm and only batchnorm benefits sharpness-aware minimization. In *Has it Trained Yet? NeurIPS 2022 Workshop*, 2022.

Möllenhoff, T. and Khan, M. E. SAM as an optimal relaxation of bayes. In *The Eleventh International Conference on Learning Representations*, 2023.

Nadiradze, G., Sabour, A., Davies, P., Li, S., and Alistarh, D. Asynchronous decentralized sgd with quantized and local updates. *Advances in Neural Information Processing Systems*, 2021.

Narayanan, D., Shoeybi, M., Casper, J., LeGresley, P., Patwary, M., Korthikanti, V., Vainbrand, D., Kashinkunti, P., Bernauer, J., Catanzaro, B., et al. Efficient large-scale language model training on gpu clusters using megatron-lm. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 1–15, 2021.

Nedic, A. Distributed gradient methods for convex machine learning problems in networks: Distributed optimization. *IEEE Signal Processing Magazine*, 2020.

Nedić, A. and Olshevsky, A. Distributed optimization over time-varying directed graphs. *IEEE Transactions on Automatic Control*, 60(3):601–615, 2014.

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

Orvieto, A., Kersting, H., Proske, F., Bach, F., and Lucchi, A. Anticorrelated noise injection for improved generalization. In *International Conference on Machine Learning*. PMLR, 2022.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 2019.

Reddi, S. J., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečný, J., Kumar, S., and McMahan, H. B. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021.

Rissanen, J. A universal prior for integers and estimation by minimum description length. *The Annals of statistics*, 11(2):416–431, 1983.

Robbins, H. E. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

Rockafellar, R. T. and Wets, R. J.-B. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

Rudin, W. et al. *Principles of mathematical analysis*. McGraw-hill New York, 1976.

Seneta, E. *Non-negative matrices and Markov chains*. Springer Science & Business Media, 2006.

Shallue, C. J., Lee, J., Antognini, J., Sohl-Dickstein, J., Frostig, R., and Dahl, G. E. Measuring the effects of data parallelism on neural network training. *Journal of Machine Learning Research*, 20(112):1–49, 2019.

Shamir, O. and Srebro, N. Distributed stochastic optimization and learning. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2014.

Shen, L., Sun, Y., Yu, Z., Ding, L., Tian, X., and Tao, D. On efficient training of large-scale deep learning models: A literature review. *arXiv preprint arXiv:2304.03589*, 2023.

Shi, W., Ling, Q., Wu, G., and Yin, W. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 2015.

Smith, L. N. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 464–472. IEEE, 2017.

Smith, S., Elsen, E., and De, S. On the generalization benefit of noise in stochastic gradient descent. In *International Conference on Machine Learning*. PMLR, 2020.

Smith, S. L., Dherin, B., Barrett, D., and De, S. On the origin of implicit regularization in stochastic gradient descent. In *International Conference on Learning Representations*, 2021.

Song, Z., Li, W., Jin, K., Shi, L., Yan, M., Yin, W., and Yuan, K. Communication-efficient topologies for decentralized learning with $o(1)$ consensus rate. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022.

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

Taheri, H., Mokhtari, A., Hassani, H., and Pedarsani, R. Quantized decentralized stochastic learning over directed graphs. In *International Conference on Machine Learning*. PMLR, 2020.

Tang, H., Lian, X., Yan, M., Zhang, C., and Liu, J. D2: Decentralized training over decentralized data. In *International Conference on Machine Learning*. PMLR, 2018.

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

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

Tsuzuku, Y., Sato, I., and Sugiyama, M. Normalized flat minima: Exploring scale invariant definition of flat minima for neural networks using pac-bayesian analysis. In *International Conference on Machine Learning*, pp. 9636–9647. PMLR, 2020.

Ujváry, S., Telek, Z., Kerekes, A., Mészáros, A., and Huszár, F. Rethinking sharpness-aware minimization as variational inference. In *OPT 2022: Optimization for Machine Learning (NeurIPS 2022 Workshop)*, 2022.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Vogels, T., He, L., Koloskova, A., Karimireddy, S. P., Lin, T., Stich, S. U., and Jaggi, M. Relaysum for decentralized deep learning on heterogeneous data. *Advances in Neural Information Processing Systems*, 34:28004–28015, 2021.

Von Neumann, J. *Some matrix-inequalities and metrization of matric space*. 1937.

Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

Wang, J., Das, R., Joshi, G., Kale, S., Xu, Z., and Zhang, T. On the unreasonable effectiveness of federated averaging with heterogeneous data. *arXiv preprint arXiv:2206.04723*, 2022.

Wang, Z., Lee, J., and Lei, Q. Reconstructing training data from model gradient, provably. In *International Conference on Artificial Intelligence and Statistics*, pp. 6595–6612. PMLR, 2023.

Warnat-Herresthal, S., Schultze, H., Shastry, K. L., Manamohan, S., Mukherjee, S., Garg, V., Sarveswara, R., Händler, K., Pickkers, P., Aziz, N. A., et al. Swarm learning for decentralized and confidential clinical machine learning. *Nature*, 2021.

Wen, K., Ma, T., and Li, Z. How sharpness-aware minimization minimizes sharpness? In *International Conference on Learning Representations*, 2023.

Wu, D., Xia, S.-T., and Wang, Y. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

Wu, L. and Su, W. J. The implicit regularization of dynamical stability in stochastic gradient descent. In *International Conference on Machine Learning*. PMLR, 2023.

Xiao, L. and Boyd, S. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 2004.

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

Yang, J., Sun, S., and Roy, D. M. Fast-rate pac-bayes generalization bounds via shifted rademacher processes. *Advances in Neural Information Processing Systems*, 32, 2019.

Yang, Z., Gang, A., and Bajwa, W. U. Adversary-resilient distributed and decentralized statistical inference and machine learning: An overview of recent advances under the byzantine threat model. *IEEE Signal Processing Magazine*, 37(3):146–159, 2020.

Yin, D., Pananjady, A., Lam, M., Papailiopoulos, D., Ramchandran, K., and Bartlett, P. Gradient diversity: a key ingredient for scalable distributed learning. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, pp. 1998–2007. PMLR, 2018.

Yin, H., Mallya, A., Vahdat, A., Alvarez, J. M., Kautz, J., and Molchanov, P. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Ying, B., Yuan, K., Chen, Y., Hu, H., Pan, P., and Yin, W. Exponential graph is provably efficient for decentralized deep training. In *Advances in Neural Information Processing Systems*, 2021a.

Ying, B., Yuan, K., Hu, H., Chen, Y., and Yin, W. Bluefog: Make decentralized algorithms practical for optimization and deep learning. *arXiv preprint arXiv:2111.04287*, 2021b.

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

You, Y., Zhang, Z., Hsieh, C.-J., Demmel, J., and Keutzer, K. Imagenet training in minutes. In *Proceedings of the 47th International Conference on Parallel Processing*. Association for Computing Machinery, 2018.

You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., and Hsieh, C.-J. Large batch optimization for deep learning: Training bert in 76 minutes. In *International Conference on Learning Representations*, 2020.

Yuan, B., He, Y., Davis, J. Q., Zhang, T., Dao, T., Chen, B., Liang, P., Re, C., and Zhang, C. Decentralized training of foundation models in heterogeneous environments. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022.

Yuan, K., Chen, Y., Huang, X., Zhang, Y., Pan, P., Xu, Y., and Yin, W. Decentlam: Decentralized momentum sgd for large-batch deep training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3029–3039, 2021.

Zellner, A. Optimal information processing and bayes's theorem. *The American Statistician*, 42(4):278–280, 1988.

Zhang, W., Cui, X., Kayi, A., Liu, M., Finkler, U., Kingsbury, B., Saon, G., Mroueh, Y., Buyuktosunoglu, A., Das, P., Kung, D. S., and Picheny, M. Improving efficiency in large-scale decentralized distributed training. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3022–3026, 2020.

Zhang, W., Liu, M., Feng, Y., Cui, X., Kingsbury, B., and Tu, Y. Loss landscape dependent self-adjusting learning rates in decentralized stochastic gradient descent. *arXiv preprint arXiv:2112.01433*, 2021.

Zheng, Y., Zhang, R., and Mao, Y. Regularizing neural networks via adversarial model perturbation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8156–8165, 2021.

Zhu, L., Liu, Z., and Han, S. Deep leakage from gradients. In *Advances in Neural Information Processing Systems*, 2019a.

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

Zhu, Z., Wu, J., Yu, B., Wu, L., and Ma, J. The anisotropic noise in stochastic gradient descent: Its behavior of escaping from sharp minima and regularization effects. In *International Conference on Machine Learning*. PMLR, 2019b.

Zhuang, J., Gong, B., Yuan, L., Cui, Y., Adam, H., Dvornek, N. C., sekhar tatikonda, s Duncan, J., and Liu, T. Surrogate gap minimization improves sharpness-aware training. In *International Conference on Learning Representations*, 2022.

A. Background

A.1. Distributed centralized learning (data decentralization)

Efficiently training large-scale models on massive amounts of data is challenging yet important for real-world applications (Narayanan et al., 2021; Shen et al., 2023). To handle an increasing amount of data and parameters, distributed learning on multiple workers emerges. Traditional distributed learning systems usually follow a centralized setup (parameter server).

Distributed centralized stochastic gradient descent (C-SGD). In C-SGD (Dean et al., 2012; Li et al., 2014), the de facto distributed training algorithm, there is only one centralized model $\mathbf{w}_{a(t)}$. C-SGD updates the model by

$$\mathbf{w}_{a(t+1)} = \mathbf{w}_{a(t)} - \eta \cdot \frac{1}{m} \sum_{j=1}^{m} \overbrace{\nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_{a(t)})}^{\text{Local gradient computation}}, \tag{A.1}$$

where η is the learning rate, $\mu_{j(t)} = \{z_{j,1}, \ldots, z_{j,|\mu_j(t)|}\}$ denotes the local training batch independent and identically distributed (i.i.d.) drawn from the data distribution $\mathcal{D}$ at the t-th iteration, and $\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu_j(t)} = \nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}) = \frac{1}{|\mu_j(t)|} \sum_{\zeta(t)=1}^{|\mu_j(t)|} \nabla \boldsymbol{L}(\mathbf{w}; z_{j,\zeta(t)})$ stacks for the local mini-batch gradient of $\boldsymbol{L}$ w.r.t. the first argument $\mathbf{w}$. The total batch size of C-SGD at t-th iteration is $|\mu(t)| = \sum_{j=1}^{m} |\mu_j(t)|$.

"Centralized" refers to the fact that in C-SGD, there is a central server receiving weight or gradient information from local workers (see Figure 2). To clarify, the process of updating the global model with the globally averaged gradients, as shown in equation A.1, is equivalent to averaging the local weights after performing local gradient updates on the global model:

$$\mathbf{w}_{j(t+1)} = \mathbf{w}_{j(t)} - \eta \cdot \nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_{a(t)}), \quad \mathbf{w}_{a(t+1)} = \frac{1}{m} \sum_{j=1}^{m} \mathbf{w}_{j(t+1)},$$

$$\Rightarrow \mathbf{w}_{a(t+1)} = \mathbf{w}_{a(t)} - \eta \cdot \frac{1}{m} \sum_{j=1}^{m} \overbrace{\nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_{a(t)})}^{\text{Local gradient computation}}. \tag{A.2}$$

C-SGD defined in Equation (A.1) and Equation (A.2) are mathematically identical to the Federated Averaging algorithm (McMahan et al., 2017) under the condition that the local step is set as 1 and all local workers are selected by the server in each round. To avoid misunderstandings, we include the term "distributed" in C-SGD to differentiate it from traditional single-worker SGD (Cauchy et al., 1847; Robbins, 1951).

A.2. Decentralized learning (data and model decentralization)

Limitations of server-based learning. Despite convenience and scalability, central server-based learning scheme suffers from two main issues: (1) a centralized communication protocol slows down training since central servers are easily overloaded, especially in low-bandwidth or high-latency cases (Lian et al., 2017); (2) there exists a potential information leakage through privacy attacks on the gradients transmitted to central server despite decentralizing data using Federated Learning (Zhu et al., 2019a; Geiping et al., 2020; Yin et al., 2021; Wang et al., 2023). As an alternative, decentralized learning (machine learning in a peer-to-peer architecture) allows workers to balance the load on the central server (Lian et al., 2017), as well as maintain confidentiality (Warnat-Herresthal et al., 2021).

Development of decentralized algorithms. The earliest work of classical decentralized optimization can be traced back to Tsitsiklis (1984), Tsitsiklis et al. (1986) and Nedic & Ozdaglar (2009). Decentralized SGD, a direct combination of decentralization and gradient-based optimization, has been extended to various contexts, including time-varying topologies (Nedić & Olshevsky, 2014; Lu & Wu, 2020; Koloskova et al., 2020; Ying et al., 2021a), directed topologies (Assran et al., 2019; Taheri et al., 2020; Song et al., 2022), asynchronous settings (Lian et al., 2018; Xu et al., 2021; Nadiradze et al., 2021; Bornstein et al., 2023), data-heterogeneous scenarios (Tang et al., 2018; Vogels et al., 2021; Le Bars et al., 2023) and Byzantine-robust versions (Yang et al., 2020; Farhadkhani et al., 2023). Although our theory focuses primarily on the vanilla decentralized SGD, we anticipate our theoretical insights will be applicable to a broad range of decentralized algorithms.

We then summarize some commonly used notions regarding decentralized learning in the following.

Definition A.1 (Doubly Stochastic Matrix)**.** *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ stand for the decentralized communication topology, where $\mathcal{V}$*

denotes the set of m computational nodes and $\mathcal{E}$ represents the edge set. For any given topology $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the doubly stochastic gossip matrix $\mathbf{P} = [\mathbf{P}_{j,k}] \in \mathbb{R}^{m \times m}$ is defined on the edge set $\mathcal{E}$ that satisfies

- $\mathbf{P} = \mathbf{P}^\top$ *(symmetric);*
- *If $j \neq k$ and $(j,k) \notin \mathcal{E}$, then $\mathbf{P}_{j,k} = 0$ (disconnected) and otherwise, $\mathbf{P}_{j,k} > 0$ (connected);*
- $\mathbf{P}_{j,k} \in [0,1]\ \ \forall k, l$ *and* $\sum_k \mathbf{P}_{j,k} = \sum_l \mathbf{P}_{j,k} = 1$ *(standard weight matrix for undirected graph).*

The doubly stochasticity of the gossip matrices is a standard assumption of decentralized learning (Lian et al., 2017; Koloskova et al., 2020). It is worth noticing that our theory is generally applicable to **arbitrary communication topologies** whose gossip matrices are doubly stochastic.

In the following we illustrate some commonly-used communication topologies.

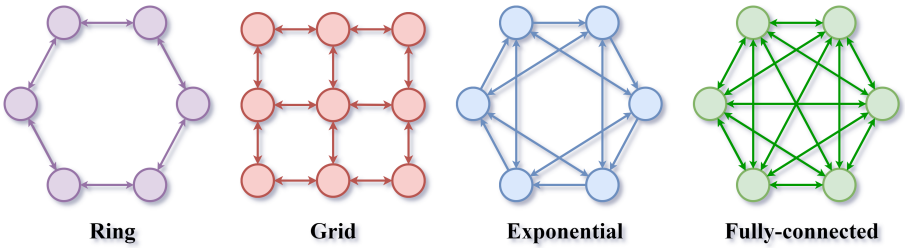


Figure A.1. An illustration of some commonly used topologies.

The intensity of gossip communications is measured by the spectral gap (Seneta, 2006) of $\mathbf{P}$.

Definition A.2 (Spectral Gap)**.** *Denote $\lambda = \max\{|\lambda_2|, |\lambda_m|\}$ where λ_i $(i = 2, \ldots, m)$ is the i-th largest eigenvalue of gossip matrix $\mathbf{P} \in \mathbb{R}^{m \times m}$. The spectral gap of a gossip matrix $\mathbf{P}$ can be defined as follows:*

$$\text{spectral gap} := 1 - \lambda.$$

According to the definition of doubly stochastic matrix (Definition A.1), we have $0 \leq \lambda < 1$. The spectral gap measures the connectivity of the communication topology. A topology is considered sparse if its communication matrix has a small spectral gap close to 0, while a topology is considered dense if its communication matrix has a large spectral gap close to 1.

A.3. Sharpness-aware minimization

Sharpness-aware minimization (SAM) is proposed by Foret et al. (2021) to minimize a perturbed loss function for the purpose of improving generalization, which is studied concurrently by Wu et al. (2020) and Zheng et al. (2021). Subsequently, various SAM variants emerge, including adaptive SAM (Kwon et al., 2021), surrogate gap guided SAM (Zhuang et al., 2022), LookSAM (Liu et al., 2022a), Fisher SAM (Kim et al., 2022b), random SAM (Liu et al., 2022b), sparse SAM (Mi et al., 2022), variational SAM (Ujváry et al., 2022) and Bayes SAM (Möllenhoff & Khan, 2023).

Definition A.3 (Vanilla SAM (Foret et al., 2021))**.** *The loss function of vanilla SAM is defined as follows:*

$$\boldsymbol{L}^{SAM}(\mathbf{w}) = \max_{\|\in\|_p \leq \rho} \boldsymbol{L}(\mathbf{w} + \epsilon).$$

Foret et al. (2021) propose to use a first-order approximation to simplify the max step:

$$\boldsymbol{L}^{SAM}(\mathbf{w}) \approx \max_{\|\in\|_p \leq \rho} [\boldsymbol{L}(\mathbf{w}) + \epsilon^\top \nabla \boldsymbol{L}(\mathbf{w})],$$

where $\epsilon^* = \rho \frac{\nabla \boldsymbol{L}(\mathbf{w})}{\|\nabla \boldsymbol{L}(\mathbf{w})\|_2}$ is the close-form solution.

Therefore, the gradient update of vanilla SAM becomes

$$\nabla \boldsymbol{L}^{SAM}(\mathbf{w}) \approx \nabla \boldsymbol{L}(\mathbf{w} + \epsilon^*) = \nabla \boldsymbol{L}(\mathbf{w} + \rho \frac{\nabla \boldsymbol{L}(\mathbf{w})}{\|\nabla \boldsymbol{L}(\mathbf{w})\|_2}).$$

Definition A.4 (Average-direction SAM (Wen et al., 2023))**.** *Average-direction SAM (AD-SAM) minimizes*

$$\boldsymbol{L}^{\text{AD-SAM}}(\mathbf{w}) = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}[\boldsymbol{L}(\mathbf{w} + \epsilon^\top \frac{\nabla \boldsymbol{L}(\mathbf{w})}{\|\nabla \boldsymbol{L}(\mathbf{w})\|_2})].$$

Average-direction is named because it minimizes an averaged loss in a "basin" around $\mathbf{w}$, rather than the original point-loss. Actually, the generalization bound of vanilla SAM in Foret et al. (2021) upper bounds the generalization error by the average-direction sharpness $\boldsymbol{L}^{\text{AD-SAM}}(\mathbf{w}) - \boldsymbol{L}(\mathbf{w})$. However, Wen et al. (2023) proves that average-direction SAM actually minimizes $\boldsymbol{L}(\mathbf{w}) + \mathrm{Tr}(\boldsymbol{H}(\mathbf{w}))$ rather than $\max_{\|\in\|_p\leq\rho} \boldsymbol{L}(\mathbf{w}+\epsilon)$.

The average-direction SAM that defined in our paper is sightly different from that in (Wen et al., 2023):

$$\underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon} - \boldsymbol{L}_{\mathbf{w}}]}_{\textit{average-direction sharpness}}.$$

This kind of AD-SAM depends both on the local landscape and consensus, which is discussed in detail in Subsection 4.1.

A.4. Generalization in large-batch training

Large-batch training is of significant interest for deep learning deployment, which can contribute to a significant speed-up in training neural networks (Goyal et al., 2017; You et al., 2018; Shallue et al., 2019). Unfortunately, it is widely observed that in the centralized learning setting, large-batch training often suffers from a drastic generalization degradation, even with fine-tuned hyperparameters, from both empirical (Chen & Huo, 2016; Keskar et al., 2017; Hoffer et al., 2017; Shallue et al., 2019; Smith et al., 2020; 2021) and theoretical (He et al., 2019; Li et al., 2021) aspects. An explanation of this phenomenon is that large-batch training lacks sufficient gradient noise to escape "sharp" minima (Smith et al., 2020).

Mitigating generalization issues in large-batch training. Linear scaling rule (LSR) is a widely used hyper-parameter-free rule to make up the noise in large-batch deep learning (He et al., 2016a; Goyal et al., 2017; Bottou et al., 2018; Smith et al., 2020), which states that a fixed learning rate to total batch size ratio allows maintaining generalization performance when the total batch size increases. Apart from LSR, various optimization techniques have been proposed to reduce the gap, including learning rate warmup (Smith, 2017), Layer-wise Adaptive Rate Scaling (LARS) (You et al., 2017) and Layer-wise Adaptive Moments (LAMB) (You et al., 2020). Is worth noticing that decentralization could be a general training technique in a data-centric setup, which can be combined with these approaches to further improve generalization in large-batch training.

A.5. β-smoothness

Smoothness is a fundamental property of the objective function in optimization, which characterizes the behavior of its gradient with respect to changes in the parameters (Boyd et al., 2004). In particular, β-smoothness quantifies the rate at which the objective function varies with respect to changes in the input variables, which is demonstrated as follow:

Definition A.5 (β-smoothness)**.** *$\boldsymbol{L}$ is β-smooth if for any z and $\mathbf{w}, \widetilde{\mathbf{w}} \in \mathbb{R}^d$,*

$$\left\|\nabla \boldsymbol{L}(\mathbf{w}; z) - \nabla \boldsymbol{L}(\widetilde{\mathbf{w}}; z)\right\|_2 \leq \beta\|\mathbf{w} - \widetilde{\mathbf{w}}\|_2. \tag{A.3}$$

An objective function that lacks β-smoothness (i.e., non-β-smoothness) can exhibit sudden variations in its gradient with respect to the input variables, making it difficult to predict the behavior of the function and to design efficient optimization algorithms. However, β-smoothness is generally difficult to ensure at the beginning and intermediate phases of deep neural network training (Bassily et al., 2020). It is noteworthy that our theoretical framework does not rely on the β-smoothness assumption of the objective function $\boldsymbol{L}$, which renders it suited for different stages of deep neural network training.

A.6. Explanation of tensor product

The tensor product between a third-order tensor $\boldsymbol{T} \in \mathbb{R}^{d\times d\times d}$ and a second-order tensor (matrix) $\boldsymbol{M} \in \mathbb{R}^{d\times d}$ in the proof of Theorem 1 is defined as

$$\underbrace{(\boldsymbol{T} \otimes \boldsymbol{M})_i}_{\text{the } i\text{-th entry}} = \text{grandsum}(\boldsymbol{T}_i \odot \boldsymbol{M}),$$

where $\boldsymbol{T}_i \in \mathbb{R}^{d\times d}$ is a second-order tensor (matrix), $\odot$ denotes the Hadamard product (Davis, 1962), and the grandsum$(\cdot)$ (Merikoski, 1984) of a second-order tensor (matrix) $\tilde{\boldsymbol{M}}$ satisfies grandsum$(\tilde{\boldsymbol{M}}) = \sum_{i,j} \tilde{\boldsymbol{M}}_{ij}$.

B. Experimental Setup and Additional Results

This section provides a comprehensive account of the experimental setup, along with supplementary experiments comparing the generalization performance and the sharpness of the minima of C-SGD and D-SGD. The validation accuracy comparison of C-SGD and D-SGD includes three scenarios: on multiple communication topologies, without pretraining, and using layer-wise learning rate tuning.

Dataset and architecture. Decentralized learning is simulated in a dataset-centric setup by uniformly partitioning data among multiple workers (GPUs) to accelerate training. Vanilla D-SGD with various commonly used topologies (see Figure 2) and C-SGD are employed to train image classifiers on CIFAR-10 (Krizhevsky et al., 2009) and Tiny ImageNet (Le & Yang, 2015) with AlexNet (Krizhevsky et al., 2017), ResNet-18 (He et al., 2016b) and DenseNet-121 (Huang et al., 2017). The ImageNet pretrained models are used as initializations to achieve better final validation performance.

Implementation setting. The number of workers (one GPU as a worker) is set as 16; and the local batch size is set as 8, 64, and 512 per worker in different cases. For the case of local batch size 64, the initial learning rate is set as 0.1 for ResNet-18 and ResNet-34 and 0.01 for AlexNet, following the setup in (Zhang et al., 2021). The learning rate is divided by 10 when the model has passed the $2/5$ and $4/5$ of the total number of iterations (He et al., 2016a). We apply the learning rate warm-up (Smith, 2017) and the linear scaling law (He et al., 2016a; Goyal et al., 2017) to maintain generalization performance with increased total batch size. Batch normalization (Ioffe & Szegedy, 2015) is employed in training AlexNet. In order to understand the effect of decentralization on the flatness of minima and generalization, all other training techniques are strictly controlled. The training accuracy is almost 100% everywhere. Exponential moving average is employed to smooth the validation accuracy curves in Figure 1, Figure B.1 and Figure B.2. The code is based on PyTorch (Paszke et al., 2019).

Hardware environment. The experiments are conducted on a computing facility with NVIDIA® Tesla™ V100 16GB GPUs and Intel® Xeon® Gold 6140 CPU @ 2.30GHz CPUs.

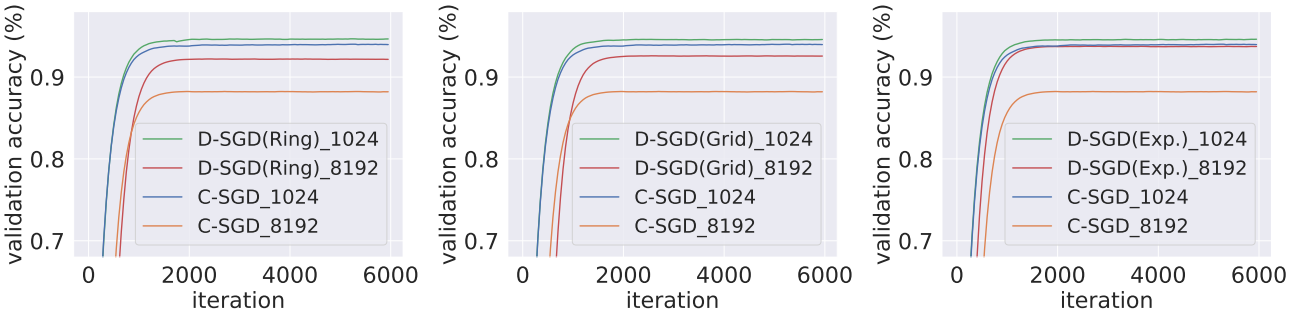


Figure B.1. The validation accuracy comparison of ResNet-18 trained on CIFAR-10 using C-SGD and D-SGD with various topologies (see Figure 2). The number of workers (one GPU as a worker) is set as 16; and the local batch size is set as 64, and 512 per worker (1024 and 8196 total batch size).

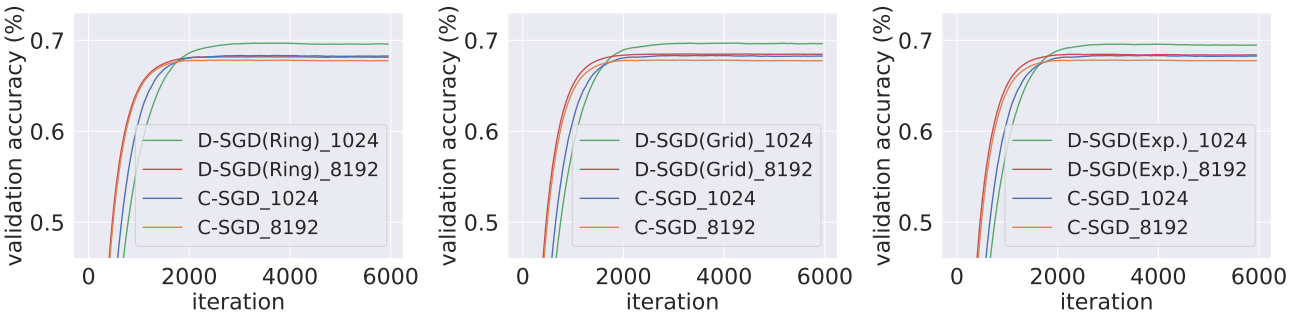
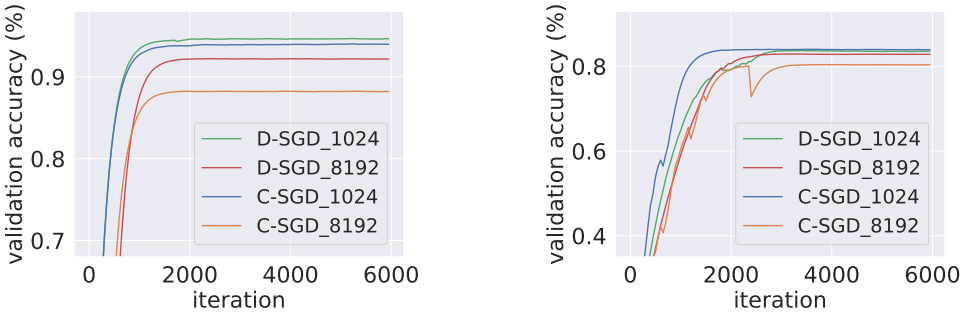


Figure B.2. The validation accuracy comparison of ResNet-18 trained on Tiny ImageNet using C-SGD and D-SGD with various topologies (see Figure 2). The number of workers (one GPU as a worker) is set as 16; and the local batch size is set as 64, and 512 per worker.

Figure B.1 and Figure B.2 show that D-SGD with different topologies consistently outperform C-SGD by a large margin in large-batch settings, which are consistent with the results shown in Figure 1.

Additional experiments without pretraining. Additional experiments are conducted to further investigate the impact of pretraining. All other training settings are kept the same as the previous experiments.



(a) ResNet-18 (with ImageNet pretrained weights) (b) ResNet-18 (training from scratch)

Figure B.3. The validation accuracy comparison of ResNet-18 with (left) and without pretraining (right) trained on CIFAR-10 using C-SGD and D-SGD. The number of workers (one GPU as a worker) is set as 16; and the local batch size is set as 64, and 512 per worker.

One can observe from Figure B.3 that the gap in validation accuracy between C-SGD and D-SGD is alleviated in the training-from-scratch settings, which we attributes to the initial optimization difficulties of decentralized training without pretrained weights (see the discussion in Section 6). The findings align with our insight that the generalization benefits of decentralization are more pronounced when optimization is not a significant obstacle. We also note that without pretraining, the accuracy curves of D-SGD are notably smoother than that of C-SGD, which supports the theoretical results in Corollary 2.

Additional experiments on LAMB. Large-batch training is of significant interest for deep learning deployment. However, linearly scaling the learning rate with the batch size can lead to generalization degradation (Shallue et al., 2019; Smith et al., 2020; 2021; Li et al., 2021). To address this issue, specialized methods have been developed to carefully tune the scaling factor between the learning rate and batch size, such as Layer-wise Adaptive Rate Scaling (LARS) (You et al., 2017) and Layer-wise Adaptive Moments (LAMB) (You et al., 2020). LARS calculates a scaling factor based on the ratio of the norm of the weight matrix to the norm of the weight gradients for each layer, while LAMB incorporates adaptive moment estimation. These methods have been shown to improve the convergence rate and generalization performance of large-batch training. We compare the validation accuracy of centralized LAMB (C-LAMB) and decentralized LAMB (D-LAMB)[12]. We follow the baseline learning rate setups (i.e., 0.0035 and 0.01) in You et al. (2020) and conduct experiments with other different learning rates, which is shown below.

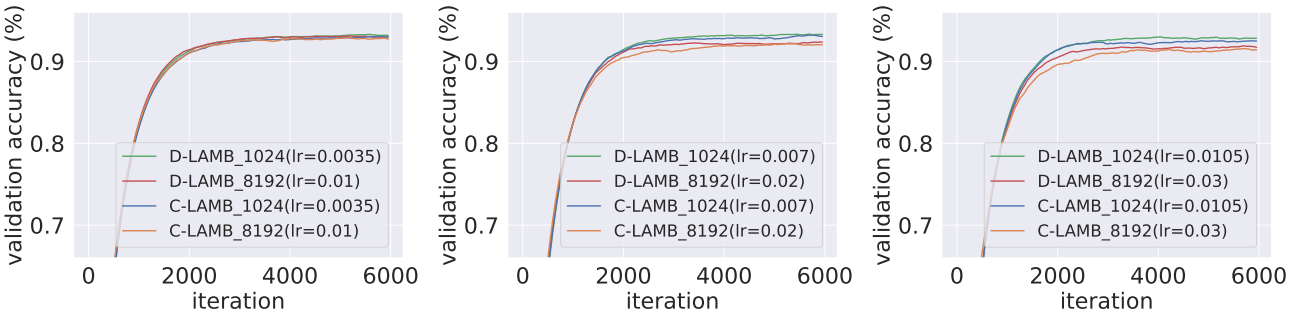


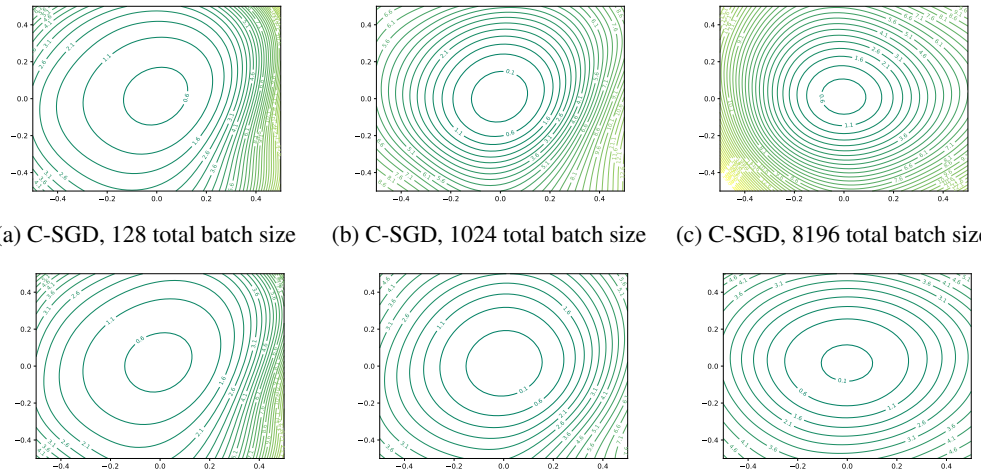
Figure B.4. The validation accuracy comparison of ResNet-18 trained on CIFAR-10 using centralized LAMB (C-LAMB) and decentralized LAMB (D-LAMB, ring topology) with different learning rates. The number of workers (one GPU as a worker) is set as 16; and the local batch size is set as 512 per worker (8196 total batch size). The baseline learning rate setups (i.e., 0.0035 and 0.01) follow You et al. (2020).

The best validation accuracy of C-LAMB training ResNet-18 on CIFAR-10 is 93.06 and 92.03 for 1024 and 8192 total batch size settings, respectively. The best validation accuracy of D-LAMB training ResNet-18 on CIFAR-10 is 93.32 and 92.95 for 1024 and 8192 total batch size settings, respectively. We find LAMB can mitigate the gap in generalizability between

[12]C-LAMB refers to the original LAMB and D-LAMB refers to the decentralized version of LAMB where the SGD optimizer in D-SGD is replaced by LAMB.

centralized training and decentralized training. However, decentralization still offers sight performance benefit on LAMB optimizer on multiple learning rates. Is is worth noticing that compared with LARS and LAMB, decentralization incurs *zero additional computation* overhead (e.g., no expensive computation of weight norm and gradient norm).

Minima visualizations. The following figures depicts the local loss landscape round minima learned by C-SGD and D-SGD.

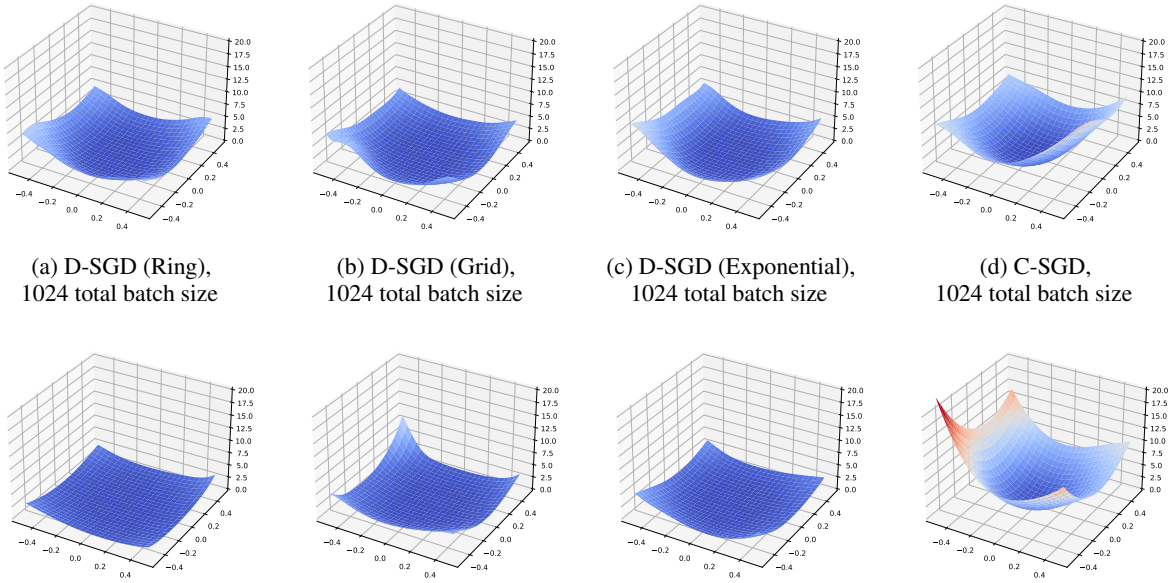


(a) C-SGD, 128 total batch size (b) C-SGD, 1024 total batch size (c) C-SGD, 8196 total batch size

(d) D-SGD, 128 total batch size (e) D-SGD, 1024 total batch size (f) D-SGD, 8196 total batch size

Figure B.5. Minima 2D visualization of ResNet-18 trained on CIFAR-10 using C-SGD and D-SGD (Ring).

We also compare the minima learned by C-SGD and D-SGD with multiple topologies in Figure B.6.



(a) D-SGD (Ring), 1024 total batch size (b) D-SGD (Grid), 1024 total batch size (c) D-SGD (Exponential), 1024 total batch size (d) C-SGD, 1024 total batch size

(e) D-SGD (Ring), 8192 total batch size (f) D-SGD (Grid), 8192 total batch size (g) D-SGD (Exponential), 8196 total batch size (h) C-SGD, 8196 total batch size

Figure B.6. Minima 3D visualization of ResNet-18 trained on CIFAR-10 using D-SGD with ring, grid-like and exponential topologies.

From Figure B.5 and Figure B.6, one can observe that (1) the minima of D-SGD with multiple commonly-used topologies are consistently flatter than those of C-SGD; and that (2) the gap in flatness increases as the total batch size increases. The findings support the claims made by Theorem 1 and Theorem 3.

C. Proof

C.1. Propositions on the consensus distance

We will introduce some useful propositions about the consensus distance, as shown below.

Proposition C.1 ((Kong et al., 2021)). *Suppose that the averaged gradient norm satisfies* $\frac{1}{m}\sum_{j=1}^{m}\left\|\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t)\right)\right\|^{2} \leq (1+\frac{1-\lambda}{4})\frac{1}{m}\sum_{j=1}^{m}\left\|\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t+1)\right)\right\|^{2}$*, then the consensus distance of D-SGD satisfies*

$$\begin{aligned}\operatorname{Tr}(\boldsymbol{\Xi}(t)) &= \frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j}(t)-\mathbf{w}_{a}(t)\|_{2}^{2} \\ &= \lambda\cdot\mathcal{O}\left(\frac{\frac{1}{m}\sum_{j=1}^{m}\left\|\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t)\right)\right\|^{2}}{(1-\lambda)^{2}}+\frac{\frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_{\mu_{j}(t)\sim\mathcal{D}}\left\|\nabla \boldsymbol{L}^{\mu_{j}(t)}\left(\mathbf{w}_{j}(t)\right)-\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t)\right)\right\|_{2}^{2}}{1-\lambda}\right),\end{aligned}$$

where λ *equals* $1-$ spectral gap *(see Definition A.2).*

Proposition C.2 (Descent condition of $\operatorname{Tr}(\boldsymbol{\Xi}(t))$)**.** *If the learning rate* η *satisfies*

$$\eta \leq \frac{\operatorname{Tr}(\boldsymbol{\Xi}(t))(1-\lambda)}{\sqrt{6}\lambda^{\frac{1}{2}}}\left[\frac{1}{m}\sum_{j=1}^{m}\left\|\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t)\right)\right\|^{2}+(1-\lambda)\cdot\frac{1}{m}\sum_{j=1}^{m}\mathbb{E}_{\mu_{j}(t)\sim\mathcal{D}}\left\|\nabla \boldsymbol{L}^{\mu_{j}(t)}\left(\mathbf{w}_{j}(t)\right)-\nabla \boldsymbol{L}\left(\mathbf{w}_{j}(t)\right)\right\|_{2}^{2}\right]^{-\frac{1}{2}},$$

it holds that,

$$\operatorname{Tr}(\boldsymbol{\Xi}(t+1)) \leq \operatorname{Tr}(\boldsymbol{\Xi}(t)).$$

As the terms in brackets are of order $\mathcal{O}\left(\frac{(1-\lambda)^{2}}{\lambda}\cdot\operatorname{Tr}(\boldsymbol{\Xi}(t))\right)$ according to Proposition C.1, the assumption on η becomes $\eta \leq \mathcal{O}(\operatorname{Tr}(\boldsymbol{\Xi}(t))^{\frac{1}{2}})$. The proof follows directly from Lemma C.2 in (Kong et al., 2021).

Proposition C.3. *It holds that,*

$$\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_{2}^{3}=(\operatorname{Tr}(\boldsymbol{\Xi}(t)))^{\frac{3}{2}}=(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j}(t)-\mathbf{w}_{a}(t)\|_{2}^{2})^{\frac{3}{2}}\leq(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_{j}(t)-\mathbf{w}_{a}(t)\|_{2}^{3}).$$

The inequality is derived from the generalized mean inequality (Marshall et al., 1979).

C.2. Proof of Theorem 1

Proposition C.4. *The gradient diversity in Equation (4) equals to zero in the following cases:*

(1) the loss is quadratic, i.e., $\boldsymbol{L}=\mathbf{w}^{\top}\boldsymbol{H}\mathbf{w}+A\mathbf{w}+b$*, where* $A\in\mathbb{R}^{d\times d}$ *and* $b\in\mathbb{R}^{d}$*;*
(2) the optimization algorithm is distributed centralized SGD (see Equation (1)).

Proof of Proposition C.4.

On quadratic loss, we have

$$\frac{1}{m}\sum_{j=1}^{m}[\nabla \boldsymbol{L}^{\mu_{j}(t)}\left(\mathbf{w}_{j}(t)\right)-\nabla \boldsymbol{L}^{\mu_{j}(t)}\left(\mathbf{w}_{a}(t)\right)]=\frac{1}{m}\sum_{j=1}^{m}[\boldsymbol{H}\mathbf{w}_{j}(t)+A-\boldsymbol{H}\mathbf{w}_{a}(t)-A]=\boldsymbol{H}\frac{1}{m}\sum_{j=1}^{m}[\mathbf{w}_{j}(t)-\mathbf{w}_{a}(t)]=0.$$

In distributed centralized SGD, the gradient diversity satisfies

$$\frac{1}{m}\sum_{j=1}^{m}[\nabla \boldsymbol{L}^{\mu_j(t)}\underbrace{(\mathbf{w}_j(t))}_{=\mathbf{w}_a(t)}-\nabla \boldsymbol{L}^{\mu_j(t)}(\mathbf{w}_a(t))]=0.$$

□

It can be deduced directly from Proposition C.4 that distributed centralized SGD, which has constant zero gradient diversity, is equivalent to standard single-worker mini-batch SGD with equivalently large batch size.

Theorem 1 (D-SGD as SAM). *Given that the loss $\boldsymbol{L}$ is continuous and has fourth-order partial derivatives. The mean iterate of the global averaged model $\mathbf{w}_a(t)=\frac{1}{m}\sum_{j=1}^{m}\mathbf{w}_j(t)$ of D-SGD can be written as follows:*

$$\mathbb{E}_{\mu(t)}[\mathbf{w}_a(t+1)]=\mathbf{w}_a(t)-\eta\underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]}_{\textit{asymptotic descent direction}}+\underbrace{\mathcal{O}(\eta\,\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_2^3+\frac{\eta}{m}\sum_{j=1}^{m}\|\mathbf{w}_j(t)-\mathbf{w}_a(t)\|_2^3)}_{\textit{higher-order residual terms}},$$

where $\boldsymbol{\Xi}(t)=\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))(\mathbf{w}_j(t)-\mathbf{w}_a(t))^T$ denotes the weight diversity matrix.

Proof of Theorem 1.

(1) Derive the iterate of the averaged model $\mathbf{w}_a(t)$.

Directly analyzing the dynamics of the diffusion-like decentralized systems where information is gradually spread across the network is non-trivial. Instead, we focus on $\mathbf{w}_a(t)=\frac{1}{m}\sum_{j=1}^{m}\mathbf{w}_j(t)$, the global averaged model of D-SGD, whose update can be written as follows,

$$\mathbf{w}_a(t+1)=\mathbf{w}_a(t)-\eta\Big[\nabla \boldsymbol{L}^{\mu(t)}_{\mathbf{w}_a(t)}+\underbrace{\frac{1}{m}\sum_{j=1}^{m}(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_j(t)}-\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_a(t)})}_{\textit{gradient diversity among local workers}}\Big]. \tag{C.1}$$

Remark. Equation (C.1) shows that decentralization introduces an additional noise, which characterizes the gradient diversity between the global averaged model $\mathbf{w}_a(t)$ and the local models $\mathbf{w}_j(t)$ for $j=1,\dots,m$, compared with its centralized counterpart. Therefore, we note that

analyzing the gradient diversity is the major challenge of decentralized (gradient-based) learning .

Insight. We also note that the gradient diversity equals to zero on quadratic objective $\boldsymbol{L}$ (see Proposition C.4). Therefore, the quadratic approximation of loss functions $\boldsymbol{L}$ (Zhu et al., 2019b; Ibayashi & Imaizumi, 2021; Liu et al., 2021; 2022c) is insufficient to characterize how decentralization impacts the training dynamics of D-SGD, especially on neural network loss landscapes where quadratic approximation may not be accurate even around minima (Ma et al., 2022). To better understand the dynamics of D-SGD on complex landscapes, it is crucial to consider *higher-order geometric information* of objective $\boldsymbol{L}$. In the following, we approximate the gradient diversity using Taylor expansion, instead of analyzing it on non-convex non-β-smooth loss $\boldsymbol{L}$ directly, which is highly non-trivial.

(2) Perform Taylor expansion on the gradient diversity. Technically, we perform a second-order Taylor expansion on the gradient diversity around $\mathbf{w}_a(t)$:

$$\frac{1}{m}\sum_{j=1}^{m}(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_j(t)}-\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_a(t)})=\frac{1}{m}\sum_{j=1}^{m}\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_a(t)}\cdot(\mathbf{w}_j(t)-\mathbf{w}_a(t))+\frac{1}{2m}\sum_{j=1}^{m}\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_a(t)}\otimes[(\mathbf{w}_j(t)-\mathbf{w}_a(t))(\mathbf{w}_j(t)-\mathbf{w}_a(t))^\top],$$

plus residual terms $\mathcal{O}(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_j(t)-\mathbf{w}_a(t)\|_2^3)$. Here $\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_a(t)}\triangleq\frac{1}{|\mu_j(t)|}\sum_{\zeta(t)=1}^{|\mu_j(t)|}\boldsymbol{H}(\mathbf{w}_a(t);z_{j,\zeta(t)})$ denotes the empirical Hessian matrix evaluated at $\mathbf{w}_a(t)$ and $\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_a(t)}\triangleq\frac{1}{|\mu_j(t)|}\sum_{\zeta(t)=1}^{|\mu_j(t)|}\boldsymbol{T}(\mathbf{w}_a(t);z_{j,\zeta(t)})$ stacks for the empirical third-order partial derivative tensor at $\mathbf{w}_a(t)$, where $\mu_j(t)$ and $z_{j,\zeta(t)}$ follows the notation in Equation (1).

As $\mathbf{w}_a(t)$ and local models $\mathbf{w}_j(t)$ $(j{=}1,\dots,m)$ are only correlated with the super batch before the t-th iteration (see Equation (2)), taking expectation over $\mu_{(t)}$ provides

$$\mathbb{E}_{\mu(t)}\big[\frac{1}{m}\sum_{j=1}^{m}(\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_j(t)}-\nabla \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_a(t)})\big] = \boldsymbol{H}_{\mathbf{w}_a(t)}\cdot\underbrace{\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))}_{=0}+\frac{1}{2}\boldsymbol{T}_{\mathbf{w}_a(t)}\otimes\big[\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))(\mathbf{w}_j(t)-\mathbf{w}_a(t))^{\top}\big],$$

plus residual terms $\mathcal{O}(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_j(t)-\mathbf{w}_a(t)\|_2^3)$, where $\boldsymbol{H}_{\mathbf{w}_a(t)}{=}\mathbb{E}_{\mu_j(t)}[\boldsymbol{H}^{\mu_j(t)}_{\mathbf{w}_a(t)}]$ and $\boldsymbol{T}_{\mathbf{w}_a(t)}{=}\mathbb{E}_{\mu_j(t)}[\boldsymbol{T}^{\mu_j(t)}_{\mathbf{w}_a(t)}]$.

The i-th entry of the above equation reads

$$\mathbb{E}_{\mu(t)}\big[\frac{1}{m}\sum_{j=1}^{m}(\partial_i \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_j(t)}-\partial_i \boldsymbol{L}^{\mu_j(t)}_{\mathbf{w}_a(t)})\big]=\frac{1}{2}\underbrace{\sum_{l,s}\partial^3_{ils}\boldsymbol{L}_{\mathbf{w}_a(t)}\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))_l(\mathbf{w}_j(t)-\mathbf{w}_a(t))_s}_{=\partial_i\sum_{ls}\partial^2_{ls}\boldsymbol{L}_{\mathbf{w}_a(t)}[\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))_l(\mathbf{w}_j(t)-\mathbf{w}_a(t))_s]}+\mathcal{O}(\frac{1}{m}\sum_{j=1}^{m}\|\mathbf{w}_j(t)-\mathbf{w}_a(t)\|_2^3), \tag{C.2}$$

where $(\mathbf{w}_j(t)-\mathbf{w}_a(t))_l$ denotes the l-th entry of the vector $\mathbf{w}_j(t)-\mathbf{w}_a(t)$. The equality in the brace is due to Clairaut's theorem (Rudin et al., 1976). The RHS of the equality is the i-th partial derivative of a quantity we term *Hessian-consensus alignment*,

$$\mathrm{Tr}(\nabla^2\boldsymbol{L}_{\mathbf{w}_a(t)}[\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))_l(\mathbf{w}_j(t)-\mathbf{w}_a(t))_s])=\sum_{ls}\partial^2_{ls}\boldsymbol{L}_{\mathbf{w}_a(t)}[\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))_l(\mathbf{w}_j(t)-\mathbf{w}_a(t))_s].$$

Denote $\boldsymbol{\Xi}(t){=}\frac{1}{m}\sum_{j=1}^{m}(\mathbf{w}_j(t)-\mathbf{w}_a(t))(\mathbf{w}_j(t)-\mathbf{w}_a(t))^T$. The positive definiteness and the non-degeneracy of $\boldsymbol{\Xi}(t)$ guarantees,

$$\begin{aligned}\mathrm{Tr}(\nabla^2\boldsymbol{L}_{\mathbf{w}_a(t)}\boldsymbol{\Xi}(t)) &= \mathrm{Tr}(\nabla^2\boldsymbol{L}_{\mathbf{w}_a(t)}\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\epsilon\epsilon^T])\\ &= \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\epsilon^T\nabla^2\boldsymbol{L}_{\mathbf{w}_a(t)}\epsilon]\\ &= \underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[2(\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}-\boldsymbol{L}_{\mathbf{w}_a(t)})}_{\text{average-direction sharpness at }\mathbf{w}_a(t)}+\mathcal{O}(\|\epsilon\|_2^3)],\end{aligned} \tag{C.3}$$

where the last equality is due to a second-order Taylor expansion of $\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}$ around $\mathbf{w}_a(t)$.

Combining Equation (C.3) and Equation (C.1) gives

$$\mathbb{E}_{\mu(t)}[\mathbf{w}_a(t{+}1)] = \mathbf{w}_a(t)-\eta\nabla\underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]}_{\text{asymptotic descent direction}}+\underbrace{\mathcal{O}(\eta\,\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_2^3+\frac{\eta}{m}\sum_{j=1}^{m}\|\mathbf{w}_j(t)-\mathbf{w}_a(t)\|_2^3)}_{\text{higher-order residual terms}},$$

which completes the proof.

□

Asymptotic equivalence. Asymptotic equivalence is a fundamental concept in mathematics which describes the equivalent behavior of functions as their inputs approach a limit point (Erdélyi, 1956; De Bruijn, 1981). Two functions are said to be asymptotically equivalent if their ratio approaches 1 as the input approaches the limit point. The asymptotic equivalence allows us to simplify complex functions and identify functions that exhibit the same limiting behavior. In the following, we define a new asymptotic equivalence called relative asymptotic equivalence, which characterizes the limiting equivalent behavior of two functions with respect to certain conditions.

Definition C.1 (Conditional asymptotic equivalence)**.** *Let $f(u_1,\cdots,u_m)$ and $g(u_1,\cdots,u_m)$ be two multivariate functions of $u_1,\cdots,u_m$, where m is an arbitrary positive integer. $f(u_1,\cdots,u_m)$ and $g(u_1,\cdots,u_m)$ are said to be conditional asymptotically equivalent with respect to the limiting condition(s) C if and only if*

$$\lim_{C}\frac{f(u_1,\cdots,u_m)}{g(u_1,\cdots,u_m)}=1. \tag{C.4}$$

The conditional asymptotic equivalence is a direct extension of the original asymptotic equivalence, where the limiting conditions are only on the original variables $u_1, \cdots, u_m$.

According to Definition C.1, $\eta \cdot \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]$ and $\mathcal{O}(\eta\, \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_2^3 + \frac{\eta}{m}\sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3)$ in Theorem 1 are conditional asymptotically equivalent with respect to the condition $(\mathbf{w}_j(t) - \mathbf{w}_a(t)) \to 0, \forall j = 1, \ldots, m$, as

$$\begin{aligned}
&\lim_{\substack{(\mathbf{w}_j(t)-\mathbf{w}_a(t))\to 0\\ \forall j=1,\ldots,m}} \frac{\eta \cdot \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}] + \mathcal{O}(\eta\, \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_2^3 + \frac{\eta}{m}\sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3)}{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]} \\
&= \lim_{\substack{(\mathbf{w}_j(t)-\mathbf{w}_a(t))\to 0\\ \forall j=1,\ldots,m}} \frac{\mu_1 \sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^2 + \mu_2 \sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3}{\mu_1 \sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^2} \\
&= 1. \qquad \text{(C.5)}
\end{aligned}$$

The first equality above is obtained by equivalent infinitesimal substitutions, where

$$\mu_1 = \lim_{\substack{(\mathbf{w}_j(t)-\mathbf{w}_a(t))\to 0\\ \forall j=1,\ldots,m}} \frac{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]}{\sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^2} \in \mathbb{R}, \text{and} \qquad \text{(C.6)}$$

$$\mu_2 = \lim_{\substack{(\mathbf{w}_j(t)-\mathbf{w}_a(t))\to 0\\ \forall j=1,\ldots,m}} \frac{\mathcal{O}(\eta\, \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\|\epsilon\|_2^3 + \frac{\eta}{m}\sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3)}{\sum_{j=1}^m \|\mathbf{w}_j(t) - \mathbf{w}_a(t)\|_2^3} \in \mathbb{R} \qquad \text{(C.7)}$$

depend only on the higher-order geometric information of $\boldsymbol{L}$ and are independent of $\mathbf{w}_j(t) - \mathbf{w}_a(t), \forall j = 1, \ldots, m$.

According to Equation (C.5), the expected gradient of D-SGD is asymptotically equivalent to $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}]$, the gradient direction of an average-direction SAM. Note that in the limit when $\mathbf{w}_j(t) - \mathbf{w}_a(t) = 0, \forall j = 1, \ldots, m$, D-SGD reduces to the standard SGD, which can be viewed as a specific instance of the average-direction SAM with $\epsilon = 0$.

C.3. Proof of Corollary 2

Corollary 2 *(Gradient smoothing effect of D-SGD). Given that vanilla loss function $\boldsymbol{L}_{\mathbf{w}}$ is α-Lipschitz continuous and the gradient $\nabla \boldsymbol{L}_{\mathbf{w}}$ is β-Lipschitz continuous. We conclude that the gradient $\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}$ where $\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}(t))$ is $\min\left\{\frac{\sqrt{2}\alpha}{\sigma_{\min}}, \beta\right\}$-Lipschitz continuous, where $\sigma_{\min}$ denotes the smallest eigenvalue of $\boldsymbol{\Xi}(t)$.*

Corollary 2 is an extension of Theorem 1 in (Liu et al., 2022b) and (Bisla et al., 2022), respectively. To prove Corollary 2, we recall Lemma 1 in (Bisla et al., 2022).

Lemma C.5 ((Bisla et al., 2022)). *Given that vanilla loss function $\boldsymbol{L}_{\mathbf{w}}$ is α-Lipschitz continuous, $\forall \mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^d$, we have*

$$\|\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\left[\nabla \boldsymbol{L}_{\mathbf{w}_1+\epsilon} - \nabla \boldsymbol{L}_{\mathbf{w}_2+\epsilon}\right]\|_2 \leq \alpha \int |p(\epsilon - \mathbf{w}_1) - p(\epsilon - \mathbf{w}_2)| d\epsilon, \qquad \text{(C.8)}$$

where p denotes the density function of $\mathcal{N}(0, \boldsymbol{\Xi}(t))$.

Proof of Corollary 2.

(1) Proving the gradient $\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}$ is $\frac{\sqrt{2}\alpha}{\sigma_{\min}}$-Lipschitz continuous.

We start by writing the RHS of Equation (C.8) as

$$\begin{aligned}
&\int |p(\epsilon - \mathbf{w}_1) - p(\epsilon - \mathbf{w}_2)| d\epsilon \leq \\
&= \int_{\epsilon:\|\epsilon-\mathbf{w}_1\|_2 \geq \|\epsilon-\mathbf{w}_2\|_2} [p(\epsilon - \mathbf{w}_1) - p(\epsilon - \mathbf{w}_2)] d\epsilon + \int_{\epsilon:\|\epsilon-\mathbf{w}_1\|_2 \leq \|\epsilon-\mathbf{w}_2\|_2} [p(\epsilon - \mathbf{w}_1) - p(\epsilon - \mathbf{w}_2)] d\epsilon \\
&= 2\int_{\epsilon:\|\epsilon-x\|_2 \leq \|\epsilon-\mathbf{w}_2\|_2} [p(\epsilon - \mathbf{w}_1) - p(\epsilon - \mathbf{w}_2)] d\epsilon.
\end{aligned}$$

The change of variables $\hat{\epsilon} = \epsilon - \mathbf{w}_1$ for $p(\epsilon - \mathbf{w}_1)$ and $\hat{\epsilon} = \epsilon - \mathbf{w}_2$ for $p(\epsilon - \mathbf{w}_2)$ gives

$$\begin{aligned}\int_{\epsilon:\|\epsilon-x\|_2\leq\|\epsilon-\mathbf{w}_2\|_2}[p(\epsilon-\mathbf{w}_1)-p(\epsilon-\mathbf{w}_2)]d\epsilon &= w\int_{\epsilon:\|\hat{\epsilon}|\leq\|\hat{\epsilon}+(x-y)\|}p(\hat{\epsilon})d\hat{\epsilon} - 2\int_{\epsilon:\|\hat{\epsilon}\|\geq\|\hat{\epsilon}-(x-y)\|}p(\hat{\epsilon})d\hat{\epsilon}\\ &= 2\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|) - 2\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\geq\|\hat{\epsilon}-(\mathbf{w}_1-\mathbf{w}_2)\|).\end{aligned}$$

The first part in the RHS of the above equality reads,

$$\begin{aligned}\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|) &= \mathbb{P}_{\hat{\epsilon}\sim p}\left(\|\hat{\epsilon}\|^2\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|^2\right)\\ &= \mathbb{P}_{\hat{\epsilon}\sim p}\left(2\langle\hat{\epsilon},\mathbf{w}_1-\mathbf{w}_2\rangle\geq-\|\mathbf{w}_1-\mathbf{w}_2\|^2\right)\\ &= \mathbb{P}_{\hat{\epsilon}\sim p}\left(2\left\langle\hat{\epsilon},\frac{\mathbf{w}_1-\mathbf{w}_2}{\|\mathbf{w}_1-\mathbf{w}_2\|}\right\rangle\geq-\|\mathbf{w}_1-\mathbf{w}_2\|\right).\end{aligned}$$

According to the fact that $\|\frac{\mathbf{w}_1-\mathbf{w}_2}{\|\mathbf{w}_1-\mathbf{w}_2\|_2}\|_2 = 1$, which implies $\left\langle\hat{\epsilon},\frac{\mathbf{w}_1-\mathbf{w}_2}{\|\mathbf{w}_1-\mathbf{w}_2\|_2}\right\rangle\sim\mathcal{N}\left(0,\frac{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}\right)$, we obtain,

$$\begin{aligned}&\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|)\\ =&\mathbb{P}_{\hat{\epsilon}\sim p}\left(\left\langle\hat{\epsilon},\frac{\mathbf{w}_1-\mathbf{w}_2}{\|\mathbf{w}_1-\mathbf{w}_2\|_2}\right\rangle\geq-\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2}\right)\\ =&\int_{-\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2}}^{+\infty}\frac{1}{\sqrt{2\pi}}\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{(\mathbf{w}_1-\mathbf{w}_2^2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\exp\left(-\frac{\hat{\epsilon}^2\|\mathbf{w}_1-\mathbf{w}_2^2\|_2}{2(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\right)d\hat{\hat{\epsilon}}.\end{aligned}$$

By some token, we have,

$$\begin{aligned}&\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}-(\mathbf{w}_1-\mathbf{w}_2)\|)\\ =&\int_{\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2}}^{+\infty}\frac{1}{\sqrt{2\pi}}\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\exp\left(-\frac{\hat{\epsilon}^2\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{2(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\right)d\hat{\hat{\epsilon}}.\end{aligned}$$

Combining the two inequalities gives

$$\begin{aligned}&\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|)\\ &=\int_{-\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2}}^{\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2}}\frac{\sqrt{2}}{\sqrt{\pi}}\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\exp\left(-\frac{\hat{\epsilon}^2\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{2(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\right)d\hat{\hat{\epsilon}}.\end{aligned}$$

Since $\exp\left(-\frac{\hat{\epsilon}^2\|\mathbf{w}_1-\mathbf{w}_2\|_2}{2(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\right)\leq 1$, we can write

$$\mathbb{P}_{\hat{\epsilon}\sim p}(\|\hat{\epsilon}\|\leq\|\hat{\epsilon}+(\mathbf{w}_1-\mathbf{w}_2)\|)\leq\frac{\sqrt{2}}{\sqrt{\pi}}\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}\|\mathbf{w}_1-\mathbf{w}_2\|_2.$$

The goal then becomes upper bounding $\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)}$ with a constant, as shown below:

$$\begin{aligned}\frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{(\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)} &= \frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{\mathrm{Tr}\left((\mathbf{w}_1-\mathbf{w}_2)^\top\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)\right)}\\ &= \frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{\mathrm{Tr}\left(\mathbf{\Xi}(t)(\mathbf{w}_1-\mathbf{w}_2)(\mathbf{w}_1-\mathbf{w}_2)^\top\right)}\\ &\leq \frac{\|\mathbf{w}_1-\mathbf{w}_2\|_2^2}{\sigma_{\min}\mathrm{Tr}\left((\mathbf{w}_1-\mathbf{w}_2)(\mathbf{w}_1-\mathbf{w}_2)^\top\right)}\end{aligned}$$

$$= \frac{1}{\sigma_{\min}},$$

where the second equality uses the cyclic property of trace and the inequality is due to Von Neumann's trace inequality (Von Neumann, 1937).

Therefore, we can prove that the gradient $\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}$ is $\frac{\sqrt{2}\alpha}{\sigma_{\min}}$-Lipschitz continuous by Lemma C.5.

(2) Proving the gradient $\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}$ is β-Lipschitz continuous.

$$\begin{aligned}
\|\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\left[\nabla \boldsymbol{L}_{\mathbf{w}_1+\epsilon} - \nabla \boldsymbol{L}_{\mathbf{w}_2+\epsilon}\right]\|_2 &\leq \| \int (\nabla \boldsymbol{L}_{\mathbf{w}_1+\epsilon} - \nabla \boldsymbol{L}_{\mathbf{w}_2+\epsilon})\, p(\epsilon)\, d\epsilon\|_2 \\
&\leq \int \|\nabla \boldsymbol{L}_{\mathbf{w}_1+\epsilon} - \nabla \boldsymbol{L}_{\mathbf{w}_2+\epsilon}\|_2\, p(\epsilon)\, d\epsilon \\
&\leq \beta\|\mathbf{w}_1 - \mathbf{w}_2\|_2 \int p(\epsilon) d\epsilon \\
&= \beta\|\mathbf{w}_1 - \mathbf{w}_2\|_2.
\end{aligned}$$

The proof is now complete.

□

C.4. Proof of Theorem C.6

Theorem C.6. *Let $\mathcal{P}$ and $\mathcal{Q}$ be the prior distribution and the posterior distribution of the model parameters, respectively. For any $\gamma \in \mathbb{R}^+$, with probability at least $1-\delta$, it holds that*

$$\underbrace{\boldsymbol{L}_{\mathbf{w}_a(t)} - \boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)}}_{\textit{generalization error}} \leq \underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)+\epsilon} - \boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)}]}_{\textit{average-direction sharpness}} + \underbrace{\boldsymbol{L}_{\mathbf{w}_a(t)} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}}_{\textit{test error perturbation}} + \mathcal{KL}(\mathcal{Q}\|\mathcal{P}) + \frac{1}{\gamma}\ln\left(\frac{1}{\delta}\right) + \frac{\gamma}{2N} \tag{C.9}$$

The test error perturbation is typically non-positive, this means that the addition of Gaussian perturbation $\epsilon \sim \mathcal{N}(0, \Sigma_q)$ does not decrease the overall test performance.

To prove Theorem C.6, we briefly introduce the PAC-Bayesian theory. The generalization performance of a model can be theoretically quantified by upper bounding the generalization gap using techniques such as the VC dimension, information theory, algorithmic stability, and PAC-Bayes. The PAC-Bayesian framework can be traced back to (McAllester, 1999), as shown in the following Lemma.

Lemma C.7 ((McAllester, 1999))**.** *Suppose the prior distribution over the parameter space is denoted by $\mathcal{P}$. Let $\mathcal{Q}$ denote the approximate posterior distribution on the parameter space represent the learned model. For any positive real $\delta \in (0,1)$, with probability at least $1-\delta$, the following inequality holds for all $\mathcal{Q}$:*

$$\underbrace{\boldsymbol{L}_{\mathbf{w}} - \boldsymbol{L}^{\mu}_{\mathbf{w}}}_{\textit{generalization gap}} \leq \sqrt{\frac{\mathcal{KL}(\mathcal{Q}\|\mathcal{P}) + \ln\frac{1}{\delta} + \ln N + 2}{2N-1}},$$

where N denotes the total sample size, $\mathcal{KL}(Q\|P)$ is the KL divergence between the distributions $\mathcal{Q}$ and $\mathcal{P}$,

$$\mathcal{KL}(\mathcal{Q}\|\mathcal{P}) = \mathbb{E}_{\mathbf{w}\sim Q}\left(\ln\frac{\mathcal{Q}(\mathbf{w})}{\mathcal{P}(\mathbf{w})}\right).$$

Some efforts, e.g., (Fawcett & Green, 2018), have been made to improve the generalization bound to

$$\underbrace{\boldsymbol{L}_{\mathbf{w}} - \boldsymbol{L}^{\mu}_{\mathbf{w}}}_{\text{generalization gap}} \leq \frac{1}{\gamma}\mathcal{KL}(\mathcal{Q}\|\mathcal{P}) + \frac{1}{\gamma}\ln(\frac{1}{\delta}) + \frac{\gamma}{2N}, \tag{C.10}$$

where γ denotes any non-negative real number.

The advantage of PAC-Bayes bounds is it can be optimized to provide useful results (Dziugaite & Roy, 2017) and is related to flatness (Yang et al., 2019; Orvieto et al., 2022). For more details on the PAC-Bayesian generalization bounds, see (Alquier, 2021).

Proof of Theorem C.6.

The proof is based on (Tsuzuku et al., 2020).

For the deterministic model $\mathbf{w}_{a(t)}$, we can rewrite Inequality (C.10) as follows:

$$\begin{aligned}\boldsymbol{L}_{\mathbf{w}_a(t)} &\leq \boldsymbol{L}_{\mathbf{w}_a(t)} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon} + \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)+\epsilon} + \frac{1}{\gamma}\mathcal{KL}(\mathcal{Q}\|\mathcal{P}) + \frac{1}{\gamma}\ln(\frac{1}{\delta}) + \frac{\gamma}{2N} \\ &\leq \boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)} + \underbrace{\boldsymbol{L}_{\mathbf{w}_a(t)} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}}_{I_1} + \underbrace{\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)+\epsilon} - \boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)}}_{I_2} + \underbrace{\frac{1}{\gamma}\mathcal{KL}(\mathcal{Q}\|\mathcal{P})}_{I_3} + \frac{1}{\gamma}\ln(\frac{1}{\delta}) + \frac{\gamma}{2N},\end{aligned} \tag{C.11}$$

where the last approximate equality omits the optimization error terms $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}^{\mu}_{\mathbf{w}_q+\epsilon} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}^{\mu}_{\mathbf{w}_a(t)+\epsilon}$ and $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_q+\epsilon}$, according to the assumption that $\mathbf{w}_{a(t)}$ is sufficiently close to $\mathbf{w}_q$. The proof is complete.

The first term $I_1 = \boldsymbol{L}_{\mathbf{w}_a(t)} - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}\boldsymbol{L}_{\mathbf{w}_a(t)+\epsilon}$ is typically non-positive, which means that adding Gaussian perturbation $\epsilon \sim \mathcal{N}(0, \boldsymbol{\Xi}_{(t)})$ does not decrease the test error and the second term is a sharpness measure.

□

C.5. Proof of Theorem 3

Theorem 4. *Let us denote $B = |\mu|$ as the total batch size. With a probability greater than $1 - \mathcal{O}(\frac{B}{(N-B)\eta^2})$, D-SGD implicit minimizes the following objective function:*

$$\boldsymbol{L}_{\mathbf{w}}^{\text{D-SGD}} \approx \boldsymbol{L}_{\mathbf{w}}^{\mu} + \underbrace{\operatorname{Tr}(\boldsymbol{H}_{\mathbf{w}}^{\mu}\boldsymbol{\Xi}(t)) + \frac{\eta}{4}\operatorname{Tr}((\boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t))}_{\text{batch size independent sharpness regularizer}} + \frac{\eta}{4}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2 + \underbrace{\frac{\kappa}{N}\sum_{j=1}^{N}\left[\|\nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2 + \operatorname{Tr}((\boldsymbol{H}_{\mathbf{w}}^{j} - \boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t))\right]}_{\text{batch size dependent variance regularizer}} + \mathcal{R}^A + \mathcal{O}(\eta^2),$$

where $\kappa = \frac{\eta}{B}\cdot\frac{N-B}{(N-1)}$, and $\mathcal{R}^A$ absorbs all higher-order residuals. The empirical loss and the gradient on the super-batch μ, denoted by $\boldsymbol{L}_{\mathbf{w}}^{\mu}$ and $\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}$, respectively, are calculated as averages over the one-sample gradients $\nabla \boldsymbol{L}_{\mathbf{w}}^{j}$. Similarly, the empirical Hessian $\boldsymbol{H}_{\mathbf{w}}^{\mu}$ is an averages of $\boldsymbol{H}_{\mathbf{w}}^{j} = \boldsymbol{H}(\mathbf{w}; z_j)$.

To prove Theorem 3, we start by showing how the gradient variance and the total batch size affect the training dynamics of the distributed central SGD.

Lemma C.8. *Recall that N denotes the total training sample size and η denotes the learning rate. Let us denote $B = |\mu|$ as the total batch size. With a probability greater than $1 - \mathcal{O}(\frac{B}{(N-B)\eta^2})$, distributed centralized SGD (see Equation (1)) implicitly minimizes the following objective function:*

$$\boldsymbol{L}_{\mathbf{w}}^{\text{C-SGD}} \approx \underbrace{\boldsymbol{L}_{\mathbf{w}}^{\mu}}_{\text{original loss}} + \frac{\eta}{4}\cdot\underbrace{\|\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2}_{\text{magnitude of gradient}} + \frac{\eta}{B}\cdot\frac{N-B}{(N-1)}\cdot\underbrace{\frac{1}{N}\sum_{j=1}^{N}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2}_{\text{variance of one-sample gradients}} + \mathcal{O}(\eta^2), \tag{C.12}$$

where the empirical loss and gradient on the super-batch μ, denoted by $\boldsymbol{L}_{\mathbf{w}}^{\mu} = \frac{1}{N}\sum_{\zeta=1}^{N}\boldsymbol{L}(\mathbf{w}; z_\zeta)$ and $\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu} = \frac{1}{N}\sum_{\zeta=1}^{N}\nabla \boldsymbol{L}(\mathbf{w}; z_\zeta)$, respectively, are calculated as averages over the one-sample gradients, represented by $\nabla \boldsymbol{L}_{\mathbf{w}}^{j} = \nabla \boldsymbol{L}(\mathbf{w}; z_j)$, at each sample z_j.

Remark. Lemma C.8 demonstrates that the "true" loss function which C-SGD (or standard SGD) optimizes is closely tracked by the original loss plus the magnitude of averaged gradient and a constant times the variance of one-sample gradients. The last term, i.e., the total variance of one-sample gradient, serves as a measure of generalizability.

Intuition of the generalization advantage of low gradient variance. To intuitively explain the regularization effect of $\frac{1}{N}\sum_{j=1}^{N}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2$, the empirical variance of gradient, we make a cartoon illustration of the loss function near two minima: the left one has low gradient variance and the right one has high gradient variance[13].

The figures depict loss functions with identical average empirical loss, yet the one-sample losses are more tightly grouped in the left figure, and dispersed in the right figure. Figure C.1 illustrates that the loss function with lower gradient variance exhibits a lower sensitivity with respect to the specific sample used for evaluation. Therefore, the minima characterized by lower gradient variance may exhibit consistency in their performance when evaluated on unseen validation samples, which guarantees good generalization performance.

Lemma C.8 proves that there exists an implicit regularization effect on the variance of the one-sample gradient in SGD to improve generalization performance. It is noteworthy that as the total batch size B approaches the total training sample size N, the regularization term diminishes rapidly, as the ratio $\frac{N-B}{N-1}$ approaches 0 asymptotically.

Proof of Lemma C.8.

[13]The illustration is inspired by a blog named "Notes on the origin of implicit regularization in SGD" written by Ferenc Huszár.

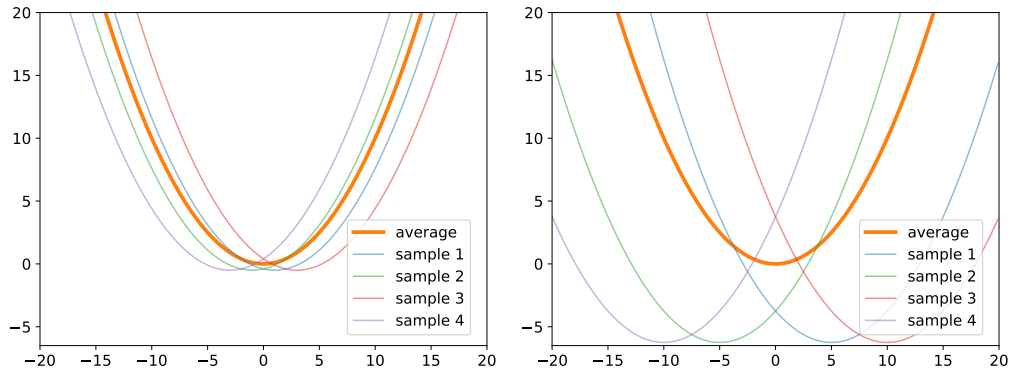


Figure C.1. An illustration of minima with low gradient variance (left) and high gradient variance (right)

We recall the true loss function approximation of single worker SGD proposed by Smith et al. (2021):

$$\boldsymbol{L}_{\mathbf{w}}^{\text{SGD}} = \boldsymbol{L}_{\mathbf{w}}^{\mu} + \frac{\eta}{4} \cdot \|\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2 + \eta \cdot \underbrace{\frac{1}{mB^2} \sum_{i=0}^{m-1} \| \sum_{j=iB+1}^{iB+B} \nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \boldsymbol{L}_{\mathbf{w}}^{\mu} \|_2^2}_{\text{variance of mini-batch gradients}}. \tag{C.13}$$

Denote $V_j = \nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}$. The total variance of the mini-batch gradient in Equation (C.13) can be written as follows:

$$\begin{aligned}\frac{1}{mB^2} \sum_{i=0}^{m-1} \| \sum_{j=iB+1}^{iB+B} V_j \|_2^2 &= \frac{1}{mB^2} \sum_{i=0}^{m-1} \sum_{j_1=iB+1}^{iB+B} \sum_{j_1=iB+1}^{iB+B} V_{j_1}^{\top} V_{j_2} \\ &= \frac{1}{mB^2} \sum_{i=0}^{m-1} \sum_{j_1=iB+1}^{iB+B} (V_{j_1}^{\top} V_{j_1} + \sum_{\substack{j_2=iB+1 \\ j_2 \neq j_1}}^{iB+B} V_{j_1}^{\top} V_{j_2}).\end{aligned}$$

The first part of the equation can be trivially written as $\frac{1}{mB^2} \sum_{i=0}^{m-1} \sum_{j_1=iB+1}^{iB+B} V_{j_1}^{\top} V_{j_1} = \frac{1}{NB} \sum_{j_1=1}^{N} V_{j_1}^{\top} V_{j_1}$. To tackle the remaining tricky part, we start by showing that

$$\frac{1}{mB^2} \sum_{i=0}^{m-1} \sum_{j_1=iB+1}^{iB+B} \sum_{\substack{j_2=iB+1 \\ j_2 \neq j_1}}^{iB+B} V_{j_1}^{\top} V_{j_2} = \frac{1}{B^2} \sum_{j_1=1}^{B} \sum_{\substack{j_2=1 \\ j_2 \neq j_1}}^{B} (\frac{1}{m} \sum_{i=0}^{m-1} V_{iB+j_1}^{\top} V_{iB+j_2}). \tag{C.14}$$

We then approximate $\frac{1}{m} \sum_{i=0}^{m-1} V_{iB+j_1}^{\top} V_{iB+j_2}$ using its expectation:

$$\frac{1}{m} \sum_{i=0}^{m-1} V_{iB+j_1}^{\top} V_{iB+j_2} = \mathbb{E}_i[V_{iB+j_1}^{\top} V_{iB+j_2}] + \underbrace{\left(\frac{1}{m} \sum_{i=0}^{m-1} V_{iB+j_1}^{\top} V_{iB+j_2} - \mathbb{E}_i[V_{iB+j_1}^{\top} V_{iB+j_2}] \right)}_{\text{bias estimator}},$$

where $\mathbb{E}_i$ eliminates the randomness of the sample z_i.

According to the Chebyshev's inequality (Chebyshev, 1874; Marshall & Olkin, 1960), with a probability greater than $1 - \frac{\sigma^2}{(m-1)\eta^2} = 1 - \frac{B\sigma^2}{(N-B)\eta^2}$ [14], it holds that,

$$\frac{1}{m} \sum_{i=0}^{m-1} V_{iB+j_1}^{\top} V_{iB+j_2} = \mathbb{E}_i[V_{iB+j_1}^{\top} V_{iB+j_2}] + \mathcal{O}(\eta),$$

[14] For sufficient large total sample size $N = \Omega(B(1+\eta^{-2}))$, the probability $1 - \frac{B\sigma^2}{N\eta^2} \approx 1$.

where σ^2 is the variance of random variables $V_{iB+j_1}^\top V_{iB+j_2}$ $(i = 1, \dots, m-1)$.

Substituting the sample mean approximation into Equation (C.14), we have that, with a probability greater than $1 - \frac{B\sigma^2}{(N-B)\eta^2}$,

$$\begin{aligned}\frac{1}{mB^2}\sum_{i=0}^{m-1}\sum_{j_1=iB+1}^{iB+B}\sum_{\substack{j_2=iB+1\\ j_2\neq j_1}}^{iB+B} V_{j_1}^\top V_{j_2} &= \frac{B(B-1)}{B^2}\left(\mathbb{E}_i[V_{iB+j_1}^\top V_{iB+j_2}(1-\delta_{j_1,j_2})] + \mathcal{O}(\eta)\right)\\ &= \frac{B(B-1)}{B^2}\left(\frac{1}{N(N-1)}\sum_{i_1=0}^{N}\sum_{\substack{i_2=0\\ i_1\neq i_2}}^{N} V_{i_1}^\top V_{i_2} + \mathcal{O}(\eta)\right),\end{aligned} \tag{C.15}$$

where δ_{j_1,j_2} denotes the Kronecker delta function. The derivation of the second equality is reminiscent of Equation (C.15), with a sight change of the indices.

Due to the fact that

$$\begin{aligned}\frac{1}{N(N-1)}\sum_{i_1=0}^{N}\sum_{i_2=0}^{N} V_{i_1}^\top V_{i_2} &= \frac{1}{N-1}\|\frac{1}{N}\sum_{i_1=1}^{N}\nabla \boldsymbol{L}_{\mathbf{w}}^{i_1} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2\\ &= \frac{1}{N-1}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2\\ &= 0,\end{aligned}$$

with a probability greater than $1 - \frac{B\sigma^2}{(N-B)\eta^2}$, the total variance of mini-batch gradient in Equation (C.13) reads,

$$\begin{aligned}\frac{1}{mB^2}\sum_{i=0}^{m-1}\|\sum_{j=iB+1}^{iB+B} V_j\|_2^2 &= \frac{1}{NB}\sum_{j_1=1}^{N} V_{j_1}^\top V_{j_1} - \frac{(B-1)}{NB(N-1)}\left(\sum_{i_1=0}^{N} V_{i_1}^\top V_{i_1}\right) + \mathcal{O}(\eta)\\ &= \frac{N-B}{(N-1)B}\cdot\underbrace{\frac{1}{N}\sum_{j=1}^{N}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2}_{\text{variance of one-sample gradients}} + \mathcal{O}(\eta).\end{aligned}$$

According to Proposition C.4, the above results also applies for distributed centralized SGD with equivalently large batch size B. The proof is now complete.

□

Theorem 3 is then established based on Lemma C.8.

Proof of Theorem 3.

According to Theorem 1, the gradient of D-SGD can be written as

$$\mathbb{E}_\mu[\nabla \boldsymbol{L}_{\mathbf{w}}^{\text{D-SGD}}] = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}] + \mathcal{R}, \tag{C.16}$$

where the higher-order residual terms are absorbed in $\mathcal{R}$ for simplicity[15].

The mini-batch form of Equation (C.16) is

$$\nabla \hat{\boldsymbol{L}}_{\mathbf{w}}^{\mu} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}^{\mu}] + \mathcal{R}^{\mu}.$$

One can also derive the corresponding gradient evaluated on the j-th sample as

$$\nabla \hat{\boldsymbol{L}}_{\mathbf{w}}^{j} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}^{j}] + \mathcal{R}^{j}.$$

[15] In the following, $\mathcal{R}^\mu$ and $\mathcal{R}^i$ denote the residual terms evaluated on super batch μ and sample j, respectively

Replacing the original loss, the mini-batch gradient $\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}$ and the j-th sample gradient $\nabla \boldsymbol{L}_{\mathbf{w}}^{j}$ in Equation (C.12) with $\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon}^{\mu}]$, $\nabla \hat{\boldsymbol{L}}_{\mathbf{w}}^{\mu}$ and $\nabla \hat{\boldsymbol{L}}_{\mathbf{w}}^{j}$, respectively, we obtain,

$$
\begin{aligned}
\boldsymbol{L}_{\mathbf{w}}^{\text{D-SGD}} =& \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\boldsymbol{L}_{\mathbf{w}+\epsilon}^{\mu}] + \frac{\eta}{4}\|\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}^{\mu}] + \nabla \mathcal{R}^{\mu}\|_2^2 + \mathcal{R}^{\mu} \\
&+ \frac{\eta}{B}\cdot\frac{N-B}{(N-1)}\cdot\frac{1}{N}\sum_{j=1}^{N}\|\mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}^{j}] - \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\boldsymbol{\Xi}(t))}[\nabla \boldsymbol{L}_{\mathbf{w}+\epsilon}^{\mu}] + \mathcal{R}^{j} - \mathcal{R}^{\mu}\|_2^2 + \mathcal{O}(\eta^2) \\
=& \boldsymbol{L}_{\mathbf{w}}^{\mu} + \operatorname{Tr}(\boldsymbol{H}_{\mathbf{w}}^{\mu}\boldsymbol{\Xi}(t)) + \frac{\eta}{4}\|\nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2 + \frac{\eta}{4}\operatorname{Tr}((\boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t)) \\
&+ \frac{\eta}{B}\cdot\frac{N-B}{(N-1)}\cdot\frac{1}{N}\sum_{j=1}^{N}\Big[\|\nabla \boldsymbol{L}_{\mathbf{w}}^{j} - \nabla \boldsymbol{L}_{\mathbf{w}}^{\mu}\|_2^2 + \operatorname{Tr}((\boldsymbol{H}_{\mathbf{w}}^{j} - \boldsymbol{H}_{\mathbf{w}}^{\mu})^2\boldsymbol{\Xi}(t))\Big] + \mathcal{R}^{A} + \mathcal{O}(\eta^2),
\end{aligned}
$$

with a probability greater than $1 - \mathcal{O}(\frac{B}{(N-B)\eta^2})$, where $\mathcal{R}^{A}$ absorb all higher-order residuals. The second equality is due to first-order Taylor expansions.

The proof is now complete.

□