Unbiased and Sign Compression in Distributed Learning: Comparing Noise Resilience via SDEs

Enea Monzio Compagnoni
Department of Mathematics
and Computer Science
University of Basel

Rustem Islamov
Department of Mathematics
and Computer Science
University of Basel

Frank Norbert Proske
Department of Mathematics
University of Oslo

Aurelien Lucchi
Department of Mathematics
and Computer Science
University of Basel

Abstract

Distributed methods are essential for handling machine learning pipelines comprising large-scale models and datasets. However, their benefits often come at the cost of increased communication overhead between the central server and agents, which can become the main bottleneck, making training costly or even unfeasible in such systems. Compression methods such as quantization and sparsification can alleviate this issue. Still, their robustness to large and heavy-tailed gradient noise, a phenomenon sometimes observed in language modeling, remains poorly understood. This work addresses this gap by analyzing Distributed Compressed SGD (DCSGD) and Distributed SignSGD (DSignSGD) using stochastic differential equations (SDEs). Our results show that DCSGD with unbiased compression is more vulnerable to noise in stochastic gradients, while DSignSGD remains robust, even under large and heavy-tailed noise. Additionally, we propose new scaling rules for hyperparameter tuning to mitigate performance degradation due to compression. These findings are empirically validated across multiple deep learning architectures and datasets, providing practical recommendations for distributed optimization.

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

1 INTRODUCTION

Recent advancements in deep learning have been fueled by the development of larger, increasingly complex models on constantly growing datasets. However, this progress comes at the expense of extended training times and resources. Therefore, distributed training has gained popularity as a way to reduce training time (27, 1). In this framework, the data is distributed among several agents or machines that collaboratively train a model being orchestrated by a server. The objective function can be expressed as an average of N functions: $\min_{x\in\mathbb{R}^d}\left[f(x) \coloneqq \frac{1}{N}\sum_{i=1}^{N} f_i(x)\right]$, where f_i represents a loss over the local data of the i-th agent, and x are the trainable parameters. Although computational resources are rapidly improving (57), the synchronization between the server and agents is still a critical performance bottleneck and can significantly increase training time (96). Among others, approaches such as communication compression (97, 3, 85), local computations (41, 61), and asynchronous communication (54, 84) are designed to boost the efficacy of distributed training.

We focus on algorithms that utilize lossy compression: They trade off some precision in the communication for reduced bandwidth usage, thereby speeding up the overall learning process. Despite the loss in precision, many compression algorithms are designed to ensure that the learning process converges to an optimal solution, often with guarantees on the convergence rate (85, 93, 36). Compression operators can be divided into two main categories: *unbiased* (e.g., sparsification (59) and quantization (47)) and *biased* (e.g., sign (12, 95), Top-k (3, 14), and low rank (107, 94, 52, 92)). While the first class is theoretically better understood

(59, 48, 85, 40, 23) and the latter is often empirically superior (97), a theoretical understanding of how these two categories differ fundamentally remains unclear, particularly regarding their behavior w.r.t. the hyperparameters of the optimizer, or their robustness to large or heavy-tailed noise.

In this work, we address these questions by comparing *unbiased* Distributed Compressed SGD (DCSGD) against Distributed SignSGD (DSignSGD), a popular *biased* compression optimizer. While the class of unbiased compressors is widely used in the literature, we specifically focus on biased *sign* compression due to its reported practical superiority (19, 64), communication efficiency (12) and connection to Adam (9). As stochastic differential equations (SDEs) have become more and more successful in offering valuable insights into the dynamics of optimization algorithms (69, 55, 75, 49, 11, 123, 25, 82, 109, 65, 8, 101, 68, 110, 10, 17, 63, 119, 105, 72, 39, 26, 81), we utilize these continuous-time models to pursue our objective. SDEs can effectively model the dynamics of discrete-time stochastic optimizers in continuous time (see Figure 1 for a graphical representation). Crucially, using SDEs allows us to leverage powerful results from Itô calculus, facilitating the derivation of convergence bounds, stationary distributions, and scaling rules *with minimal mathematical effort*. This approach is especially useful for analyzing the intricate interactions between the optimization landscape, stochastic noise, and compression. Finally, another significant advantage of SDEs is that they enable a direct comparison between optimizers by making explicit the impact of hyperparameters and landscape features on their behavior (22, 69, 76, 87).

Contributions. We identify the following as key ones:

1. We derive the first SDEs for DSGD, DCSGD, and DSignSGD under general assumptions and compare their behavior in terms of expected loss, expected gradient norm, and stationary distribution. Importantly, we discover that *sign* and *unbiased* compression interact differently with gradient noise;
2. For SignSGD, we prove that *sign* compression causes the noise level, e.g. standard deviation or scale, to inversely affect the convergence rate of both the loss and the iterates. This is in contrast with DCSGD for which the noise level plays no role. Additionally, the noise level has a linear impact on the asymptotic expected loss and covariance of the iterates while this is quadratic for DCSGD;
3. We show that heavy-tailed noise marginally affects the performance of DSignSGD, which remains robust even to noise of infinite expected value. Under the same conditions, DCSGD fails to converge;
4. We derive novel scaling rules for DCSGD and DSignSGD: These rules provide intuitive and actionable guidelines to adjust hyperparameters, e.g. to contrast the performance degradation due to compression, or adapt to newly available hardware;
5. We empirically verify every theoretical insight and prediction. Our experiments are conducted on a variety of deep learning architectures (MLP, ResNet, ViT, GPT2) and datasets (Breast Cancer Wisconsin, MNIST, CIFAR-10, FineWeb-Edu).

2 RELATED WORKS

SDE Approximations and Applications. In (69), a formal theoretical framework was proposed to derive SDEs that accurately capture the stochastic nature inherent in optimization methods commonly used in machine learning. Since then, SDEs have been applied in various areas of machine learning, including *stochastic optimal control* to optimally adjust both stepsize (69, 70) and batch size (120). Importantly, they have been used to characterize *convergence bounds* and *stationary distributions* (20, 22), *scaling rules* (55, 76), and provided insights in the context of *implicit regularization* (100, 20).

Two Classes of Compression. The current theory focuses *either* on unbiased (24, 90, 85, 53) *or* biased (36, 34) compression without discussing the conceptual differences between the two classes. However, biased compressors typically outperform their unbiased counterparts in practical applications (97). Therefore, there is a gap between theory and practice which we aim to reduce in this paper.

Heavy-tailed Noise. Recent empirical evidence suggests that the noise in several deep learning setups follows a heavy-tailed distribution (99, 118, 44, 64). In contrast, previous studies mostly focused on more restricted bounded variance assumptions. Therefore, there is a growing interest in analyzing the convergence of algorithms under heavy-tailed noise (31, 104, 115, 42). Earlier works (60, 67, 116) combined compression and clipping to make the algorithm communication-efficient and robust to heavy-tailed noise: We show that the *sign* compressor alone effectively addresses both issues without introducing additional hyperparameters.

3 FORMAL STATEMENTS & INSIGHTS THROUGH SDEs

This section provides the formulations of the SDEs of DSGD (Theorem 3.2), DCSGD (Theorem 3.6) and DSignSGD (Theorem 3.10): We use these models to derive convergence rates, scaling rules, and stationary distributions of the respective optimizers. Given the technical complexity of the analysis, the formal statements and proofs are provided in the appendix for further reference.

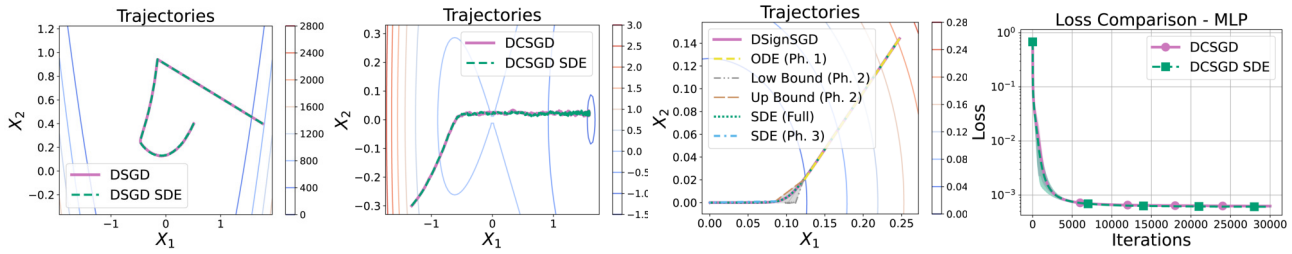

Figure 1: Empirical validation that the trajectories of the SDEs match those of the respective algorithm averaged over 500 runs - DSGD (Thm. 3.2) on a Rosenbrock function (**Left**); DCSGD (Thm. 3.6) with Rand-k on an Embedded Saddle (**Center-Left**); DSignSGD on a Convex Quadratic: As per Thm. 3.10, the dynamics of DSignSGD can be partitioned into three phases — Not only the "Full" SDE is a faithful model for DSignSGD through the whole dynamics, but so are the ODE of Ph. 1 and the SDE of Ph. 3 in their respective phases. Importantly, the bound that characterizes Ph. 2 captures the dynamics as prescribed (**Center-Right**); The SDEs and the optimizers move at the same speed — DCSGD on an MLP (**Right**). For details, see Appendix F.

Assumptions and Notation. In this section, we assume that the stochastic gradient of the i-th agent is given by $\nabla f_{\gamma_i}(x) = \nabla f(x) + Z_i(x)$, where $Z_i(x)$ denotes the gradient noise and $Z_i(x)$ is independent of $Z_j(x)$ for $i \neq j$. If $Z_i(x) \in L^1(\mathbb{R}^d)$, we assume $\mathbb{E}[Z_i(x)] = 0$, and if $Z_i(x) \in L^2(\mathbb{R}^d)$, we assume $Cov(Z_i(x)) = \Sigma_i(x)$ [1] s.t. $\sqrt{\Sigma_i(x)}$ is bounded, Lipschitz, satisfies affine growth, and together with its derivatives, it grows at most polynomially fast (Definition 2.5 in (76)). Importantly, we assume that all $Z_i(x)$ have a smooth and bounded probability density function whose derivatives are all integrable: A common assumption in the literature is for $Z_i(x)$ to be Gaussian [2] (2, 18, 77, 102, 124, 112, 114) while our assumption allows for heavy-tailed distributions such as the Student's t. To derive the stationary distribution near the optimum, we approximate the loss function as a quadratic convex function $f(x) = \frac{1}{2}x^\top Hx$, a standard approach in the literature (38, 66, 56, 91, 78, 20).

About notation, n_i is the number of data points in the local dataset of the i-th agent, $\eta > 0$ is the learning rate, and the batches $\{\gamma_k\}$ have size $B \geq 1$ and are modeled as i.i.d. random variables uniformly distributed over $\{1, \dots, n_i\}$. Finally, W_t is the Brownian motion.

The following definition formalizes in which sense an SDE can be "reliable" in modeling an optimizer.

Definition 3.1 (Weak Approximation)**.** A continuous-time stochastic process $\{X_t\}_{t\in[0,T]}$ is an order α weak approximation of a discrete stochastic process $\{x_k\}_{k=0}^{\lfloor T/\eta \rfloor}$ if for every polynomial growth function g, there exists a positive constant C, independent of the stepsize η, such that $\max_{k=0,\dots,\lfloor T/\eta \rfloor} |\mathbb{E}g(x_k) - \mathbb{E}g(X_{k\eta})| \leq C\eta^\alpha$.

Rooted in the numerical analysis of SDEs (83) this definition quantifies the discrepancy between the continuous-time model X_t and discrete-time process x_k.

3.1 Distributed SGD

In this section, we derive an SDE model for Distributed SGD whose update rule is

$$x_{k+1} = x_k - \frac{\eta}{N}\sum_{i=1}^{N} \nabla f_{\gamma_i}(x_k). \tag{1}$$

Though not surprising, the following results serve as a reference point for analyzing other optimizers in the subsequent sections. The first shows the SDE of DSGD which we validate in Fig. 1 on a simple landscape.

Theorem 3.2 (Informal Statement of Theorem B.8)**.** *The SDE of DSGD is*

$$dX_t = -\nabla f(X_t)dt + \sqrt{\frac{\eta}{NB}}\sqrt{\hat{\Sigma}(X_t)}dW_t, \tag{2}$$

where $\hat{\Sigma}(x) \coloneqq \frac{1}{N}\sum_{i=1}^{N}\Sigma_i(x)$ *is the average of the covariance matrices of the* N *agents.*

Effectively, the SDE above extends the single-node case presented by (69), where the batch size B is replaced by BN. Using the SDE established in Thm. 3.2, we derive the convergence rate of DSGD for smooth functions that satisfy the PL condition (58).

Theorem 3.3. *If* f *is* μ*-PL,* L*-smooth,* $Tr(\Sigma_i(x)) < \mathcal{L}_{\sigma_i}$, $\overline{\mathcal{L}}_\sigma \coloneqq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\sigma_i}$, *and* $S_t \coloneqq f(X_t) - f(X_*)$

$$\mathbb{E}[S_t] \leq S_0 e^{-2\mu t} + (1 - e^{-2\mu t})\frac{\eta L \overline{\mathcal{L}}_\sigma}{4\mu BN}. \tag{3}$$

For the general smooth non-convex functions the convergence guarantees are given in the next theorem.

Theorem 3.4. *If* f *is* L*-smooth, we use a learning rate scheduler* η_t *such that* $\phi_t^i = \int_0^t (\eta_s)^i ds$, $\phi_t^1 \overset{t\to\infty}{\to} \infty$, $\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0$, $\overline{\mathcal{L}}_\sigma \coloneqq \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\sigma_i}$, *and* $\tilde{t}$ *distributed as* $\frac{\eta_{\tilde{t}}}{\phi_t^1}$,

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right] \leq \frac{f(X_0) - f(X_*)}{\phi_t^1} + \frac{\eta L \overline{\mathcal{L}}_\sigma}{2BN}\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0. \tag{4}$$

[1] We omit the size of the batch γ unless relevant.

[2] See (55) for the justification why this might be the case.

Observations on convergence guarantees:

1. In Thm. 3.3, the decay is $e^{-2\mu t}$, as in SGD;
2. In Thm. 3.3, the asymptotic expected loss scales inversely to N, i.e. DCSGD attains a linear speedup with the number of agents N. Moreover, the stochastic term is proportional to the condition number $\frac{L}{\mu}$ of the Hessian of the loss, and scales with the average variance $\overline{\mathcal{L}}_\sigma$ of the gradient noise which is also observed in earlier works (37);
3. Analogous conclusions hold in Thm. 3.4.

Scaling Rules: Preserving DSGD Performance

After an extensive hyperparameter tuning phase of a machine learning model, it is common to need adjustments to the hyperparameters to accommodate new scenarios. For instance, when training LLMs, larger batch sizes may be desirable to fully utilize newly available and larger GPUs, without the need to repeat the fine-tuning process. Scaling rules offer theoretically grounded formulas that allow changes to some hyperparameters while adjusting others to maintain specific performance metrics. These rules are not *strict* but serve as guidelines to *narrow down* the hyperparameter search space. Common scaling rules include the *linear* rule for SGD (55) that prescribes that the ratio of learning rate and batch size must be kept constant and the far more complex *square-root* rule of Adam (76). As we demonstrate next, we extend the *linear* scaling rule of SGD to the distributed setting, thereby incorporating the number of agents. To establish this scaling rule, we seek a functional relationship between the hyperparameters, ensuring that DSGD with a learning rate of η, batch size B, and N agents achieves the same performance as with a learning rate of $\kappa\eta$, batch size δB, and αN agents. The next result shows the exact dependencies.

Proposition 3.5. *The scaling rule to preserve the performance independently of δ, κ, and α is $\frac{\kappa}{\alpha\delta} = 1$.*

3.2 Distributed Compressed SGD

Next, we study Distributed Compressed SGD whose update rule has a form as

$$x_{k+1} = x_k - \frac{\eta}{N}\sum_{i=1}^{N} \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right), \tag{5}$$

where the stochastic compressors $\mathcal{C}_{\xi_i}$ are independent for different i and satisfy (i) $\mathbb{E}_{\xi_i}\left[\mathcal{C}_{\xi_i}(x)\right] = x$ and (ii) $\mathbb{E}_{\xi_i}\left[\|\mathcal{C}_{\xi_i}(x) - x\|_2^2\right] \leq \omega_i \|x\|_2^2$ for some compression rates $\omega_i \geq 0$. The following result shows the SDE of DCSGD, which we believe to be a novel addition to the literature and reveals the unique manner in which gradient noise and *unbiased* compression influence the dynamics of DCSGD — See Fig. 1 for its validation on a simple landscape and MLP training with *Rand-k*.

Theorem 3.6 (Informal Statement of Theorem C.8)**.** *The SDE of DCSGD is*

$$dX_t = -\nabla f(X_t)dt + \sqrt{\frac{\eta}{N}}\sqrt{\tilde{\Sigma}(X_t)}dW_t, \tag{6}$$

where for $\Phi_{\xi_i,\gamma_i}(x) := \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f_{\gamma_i}(x)$

$$\tilde{\Sigma}(x) = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbb{E}_{\xi_i\gamma_i}\left[\Phi_{\xi_i,\gamma_i}(x)\Phi_{\xi_i,\gamma_i}(x)^\top\right] + \Sigma_i(x)\right). \tag{7}$$

The covariance matrix for DCSGD consists of the covariance matrix of DSGD plus an additional component due to the compression, e.g. for k-Sparsification, $\tilde{\Sigma}(x) = \left(\frac{d}{k} - 1\right)\left(\nabla f(x)\nabla f(x)^\top + \hat{\Sigma}(x)\right) + \hat{\Sigma}(x)$.

We derive convergence rates for the loss value and gradient norm by leveraging the SDE derived in Thm. 3.6: These recover the best known results in the literature (59, 73, 90), thus testifying that SDEs provide the community with a powerful alternative technique that allows for a *precise analysis* of optimizers. We start with the convergence guarantees for PL functions.

Theorem 3.7. *If f is μ-PL, L-smooth, $\overline{\omega} = \frac{\sum_{i=1}^N \omega_i}{N}$, $Tr(\Sigma_i(x)) < \mathcal{L}_{\sigma_i}$, $\overline{\mathcal{L}}_\sigma := \frac{\sum_{i=1}^N \mathcal{L}_{\sigma_i}}{N}$, $\overline{\omega\mathcal{L}_\sigma} := \frac{\sum_{i=1}^N \omega_i\mathcal{L}_{\sigma_i}}{N}$, $S_t := f(X_t) - f(X_*)$, and $\Delta := 1 - \frac{\eta L^2\overline{\omega}}{2\mu N}$, then*

$$\mathbb{E}\left[S_t\right] \leq S_0 e^{-2\mu\Delta t} + \left(1 - e^{-2\mu\Delta t}\right)\frac{\eta L\left(\overline{\mathcal{L}}_\sigma + \overline{\omega\mathcal{L}_\sigma}\right)}{4\mu BN\Delta}. \tag{8}$$

The next theorem offers a *new* and *general* condition on the learning rate scheduler to ensure the convergence of DCSGD for the general non-convex case.

Theorem 3.8. *If f is L-smooth and the learning rate scheduler η_t is such that $\phi_t^i = \int_0^t (\eta_s)^i ds$, $\phi_t^1 \overset{t\to\infty}{\to} \infty$, $\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0$, $\eta_t < \frac{2N}{\eta L\overline{\omega}}$, and $S_0 := f(X_0) - f(X_*)$ then, $\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right]$ is smaller than*

$$\frac{1}{1 - \frac{\eta L\overline{\omega}}{2N}\frac{\phi_t^2}{\phi_t^1}}\left(\frac{S_0}{\phi_t^1} + \frac{\phi_t^2}{\phi_t^1}\frac{\eta L\left(\overline{\mathcal{L}}_\sigma + \overline{\omega\mathcal{L}_\sigma}\right)}{2BN}\right) \overset{t\to\infty}{\to} 0, \tag{9}$$

where $\tilde{t}$, is a random time with distribution $\frac{\eta_{\tilde{t}} - \frac{\eta L\overline{\omega}}{2N}(\eta_{\tilde{t}})^2}{\phi_t^1 - \frac{\eta L\overline{\omega}}{2N}\phi_t^2}$.

Observations on convergence guarantees:

1. The decay $e^{-2\mu\Delta t}$ of DCSGD is strictly slower than that of DSGD: Δ crucially depends on the average rate of compression $\overline{\omega}$, the condition number, and the number of agents. Specifically, **larger compression implies a slower convergence in comparison with DSGD**, which is exacerbated for ill-conditioned landscapes;

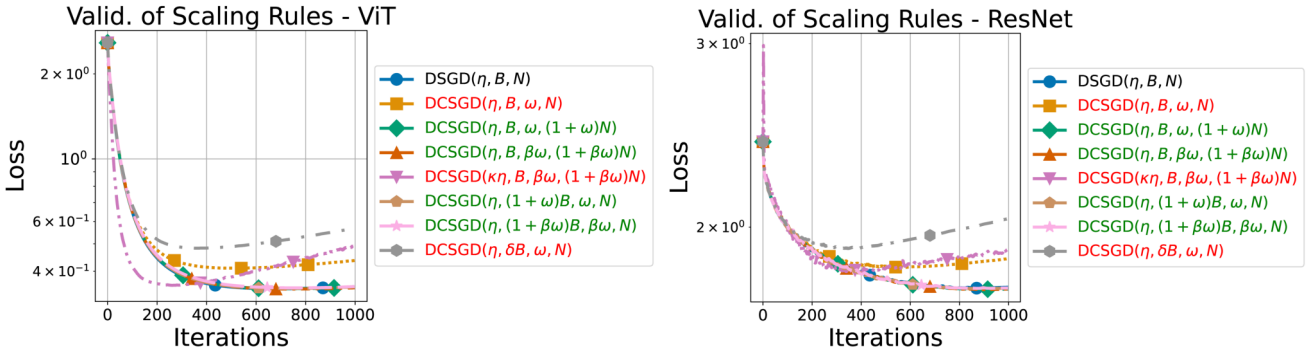

Figure 2: Validation of Scaling Rules: Consistently with Prop. 3.9, DCSGD run with hyperparameters that follow the scaling rules listed in Table 1 (marked in green in the legends) recover the performance of DSGD(η, B, N). Those that do not (marked in red) fail to do so. On the left, we plot the training loss of a ViT for *some* rules while on the right we do the same for a ResNet. Details are in Appendix F.

2. Δ needs to be positive to ensure convergence, which imposes limitations on the hyperparameters. For example, $\eta < \frac{2\mu N}{L^2\overline{\omega}}$: More agents allow for a larger learning rate, but a larger compression rate or an ill-conditioned landscape restricts it. **Violating such prescriptions might lead to divergence** (See left of Fig. 8 for empirical validation);

3. DCSGD enjoys a linear speedup: The asymptotic loss level in Thm. 3.7 **scales inversely to the number of agents** N and has an additional term $\overline{\omega\mathcal{L}_\sigma}$ w.r.t. DSGD (Thm. 3.3) which quantifies the nonlinear interaction between the compression and gradient noise. See the center-left and -right of Figure 8 for empirical validation).

Scaling Rules: Recovering DSGD Performance

As previously noted, when using the same learning rate η, batch size B, and N agents, DCSGD is slower and less optimal than DSGD. To address this, we propose deriving novel, actionable, and interpretable scaling rules to adjust the hyperparameters of DCSGD to *recover* the asymptotic performance of DSGD. The following result shows these rules under the simplifying assumption that $\mathcal{L}_{\sigma_i} = \mathcal{L}_\sigma$, $\omega_i = \omega$, for $i \in [N]$, and $N \gg 1$. We defer to Prop. C.12 for the general cases. We validate some of these rules in Figure 2 for a ViT and a ResNet. See Appendix G for the validation on a 124M GPT2 model.

Proposition 3.9. *Let DCSGD$(\kappa\eta, \delta B, \beta\omega, \alpha N)$ run with batch size δB, learning rate $\kappa\eta$, compression rates $\beta\omega$, and αN agents. Table 1 shows scaling rules to recover the asymptotic performance of DSGD(η, B, N):* [3]

[3]For practical reasons, we only show those involving two hyperparameters while the other two are kept constant.

Scaling Rule	Implication
$\alpha = 1 + \beta\omega$	CR $\uparrow \implies$ Agents $\uparrow$
$\alpha = \kappa(1 + \omega)$	LR $\uparrow \implies$ Agents $\uparrow$
$\alpha = \frac{1+\omega}{\delta}$	BS $\downarrow \implies$ Agents $\uparrow$
$\kappa = \frac{1}{1+\beta\omega}$	CR $\uparrow \implies$ LR $\downarrow$
$\delta = 1 + \beta\omega$	CR $\uparrow \implies$ BS $\uparrow$
$\kappa = \frac{\delta}{1+\omega}$	BS $\uparrow \implies$ LR $\uparrow$

Table 1: Scaling Rules to *recover* DSGD performance. For example, compression can be countered by increasing the number of agents (CR = Compression Rate, LR = Learning Rate, and BS = Batch Size).

Observations on scaling rules:

1. In the absence of compression, the scaling rules all reduce to that of DSGD (See Prop. 3.5);
2. For example, to achieve comparable performance between DSGD and DCSGD with compression factor ω, the number of agents can be *increased* to $(1 + \omega)N$. Alternatively, one can further *increase* compression ($\beta > 1$) and compensate by *increasing* the number of agents to $(1 + \beta\omega)N$. Similarly, one can *decrease* the learning rate to $\frac{\eta}{1+\omega}$, or *increase* the batch size to $B(1 + \omega)$.

3.3 Distributed SignSGD

Now we turn to derive an SDE for DSignSGD, a *biased* compression method with update rule

$$x_{k+1} = x_k - \frac{\eta}{N}\sum_{i=1}^{N} \text{sign}(\nabla f_{\gamma_i}(x_k)). \tag{10}$$

This derivation reveals how *sign* compression interacts with the gradient noise in determining the dynamics of DSignSGD. In particular, we focus on the role of the **level** of the gradient noise, e.g. standard deviation or scale, and of the **fatness** of the tails of its distribution. See (21) for a comparison between SignSGD and RMSprop, Adam, and AdamW in the single-node case.

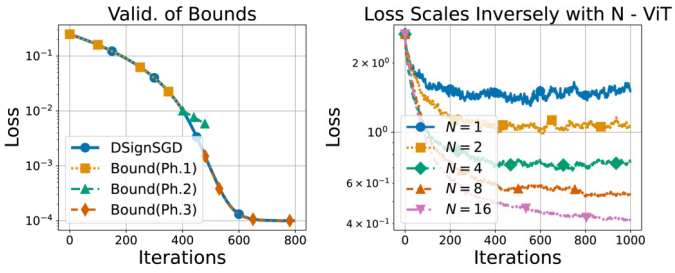

Figure 3: Validation of Bounds: As prescribed by Thm. 3.12, the bounds match or dominate the empirical loss of DSignSGD on a quadratic convex function in all three phases (**Left**); As per Thm. 3.12, DSignSGD achieves linear speedup: More agents imply lower loss (**Right**);

Theorem 3.10 (Informal Statement of Theorem D.9)**.** *The SDE of DSignSGD is*

$$dX_t = -\frac{1}{N}\sum_{i=1}^{N}\left(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(X_t) < 0)\right)dt + \sqrt{\frac{\eta}{N}}\sqrt{\overline{\Sigma}(X_t)}dW_t, \tag{11}$$

where $\overline{\Sigma}(x) := \frac{\sum_{i=1}^{N}\overline{\Sigma_i}(x)}{N}$, $\overline{\Sigma_i}(x) = \mathbb{E}[\xi_{\gamma_i}(x)\xi_{\gamma_i}(x)^\top]$, *and* $\xi_{\gamma_i}(x) := sign(\nabla f_{\gamma_i}(x)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_i}(x) < 0)$ *is the noise around* $sign\left(\nabla f_{\gamma_i}(x)\right)$.

For interpretability reasons, we present a corollary with a flexible gradient noise assumption that interpolates between the Cauchy (**heavy**-tailed) and Gaussian (**light**-tailed) distributions: $\nabla f_{\gamma_i}(x) = \nabla f(x) + \sqrt{\frac{\Sigma_i}{B}}Z_i$, $Z_i \sim t_\nu(0, I_d)$, ν are the degrees of freedom, and scale matrices $\Sigma_i = \text{diag}(\sigma_{1,i}^2, \cdots, \sigma_{d,i}^2)$. This allows us to **parse the dynamics of DSignSGD into three distinct phases** depending on the size of the signal-to-noise ratios $Y_t^i := \sqrt{B}\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)$. This is visually supported in the center-right of Fig. 1 on a convex quadratic function.

The following results involve several quantities, highlighted using colors for clarity: *Pink* indicates dependence on the degrees of freedom ν, related to the concept of *fatness* of the noise, while *blue* corresponds to the *level* of noise.

Proposition 3.11. *For some constants* $\mathbf{q}_\nu^+$, $\mathbf{q}_\nu^-$, m_ν, ℓ_ν, *and* ψ_ν *that depend on the degrees of freedom* ν, [4] *the dynamics of DSignSGD has three phases:*

1. ***Phase 1:*** *If* $\left|Y_t^i\right| > \psi_\nu$, *the SDE coincides with the ODE of SignGD:*

$$dX_t = -sign(\nabla f(X_t))dt; \tag{12}$$

2. ***Phase 2:*** *If* $1 < \left|Y_t^i\right| < \psi_\nu$ *and* $\overline{Y_t} := \frac{\sum_{i=1}^{N} Y_t^i}{N}$: $\mathbb{P}\left[\|X_t - \mathbb{E}[X_t]\|_2^2 > a\right] \le \frac{\eta}{a}\left[d - \frac{\sum_{i=1}^{N}\|m_\nu Y_t^i + \mathbf{q}_\nu^-\|_2^2}{N}\right]$;

3. ***Phase 3:*** *If* $\left|Y_t^i\right| < 1$, *the SDE is*

$$dX_t = -\ell_\nu\left(\frac{\sqrt{B}}{N}\sum_{i=1}^{N}\Sigma_i^{-\frac{1}{2}}\right)\nabla f(X_t)dt + \sqrt{\frac{\eta}{N}}\sqrt{I_d - \frac{\ell_\nu^2 B}{N}\sum_{i=1}^{N}\text{diag}\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)^2}dW_t. \tag{13}$$

Observation on SDEs:

1. The behavior of DSignSGD depends on the size of the signal-to-noise ratios;
2. In Ph. 2 and Ph. 3, the inverse of the level of the noise $\Sigma_i^{-\frac{1}{2}}$ premultiplies the gradient, thus **affecting the rate of descent**: The larger the scale, the slower the dynamics. This is not the case for DCSGD where the Σ_i **only influence the diffusion term**;
3. The degrees of freedom ν of the Student's t parametrize the fatness of the tails of the noise distribution: The smaller ν, the fatter the tails and the smaller m_ν and ℓ_ν [5] — **Fatter tails** imply that the average dynamics of X_t is **slower** and exhibits **more variance**.

To better understand the role of the noise, we need to study how its level and fatness affect the dynamics of the expected loss. The tightness of these bounds is empirically verified on the left of Fig. 3.

Theorem 3.12. *Let* f *be* μ*-strongly convex,* $Tr(\nabla^2 f(x)) < \mathcal{L}_\tau$, $\Sigma_i \le \sigma_{max,i}^2$, $S_t := f(X_t) - f(X_*)$, *and* $\sigma_{\mathcal{H},j}$ *be the harmonic mean of* $\{(\sigma_{max,i})^j\}$. *Then,*

1. *In* ***Phase 1****,* $S_t \le \frac{1}{4}\left(\sqrt{\mu}t - 2\sqrt{S_0}\right)^2$: *DSignSGD stays in this phase for at most* $t_* = 2\sqrt{\frac{S_0}{\mu}}$;
2. *In* ***Phase 2****, for* $\Delta := m_\nu\sqrt{B}\sigma_{\mathcal{H},1}^{-1} + \frac{\eta B\mu m_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}$, $\mathbb{E}[S_t] \le S_0 e^{-2\mu\Delta t} + \frac{\eta(\mathcal{L}_\tau - \mu d\hat{q}_\nu^2)}{2N}\frac{1}{2\mu\Delta}\left(1 - e^{-2\mu\Delta t}\right)$;
3. *In* ***Phase 3****, for* $\Delta := \ell_\nu\sqrt{B}\sigma_{\mathcal{H},1}^{-1} + \frac{\eta B\mu\ell_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}$, $\mathbb{E}[S_t] \le S_0 e^{-2\mu\Delta t} + \frac{\eta\mathcal{L}_\tau}{2N}\frac{1}{2\mu\Delta}\left(1 - e^{-2\mu\Delta t}\right)$.

Observations on Convergence - PL:

1. The dynamics of Ph. 1 ensures a steady decrease of S_t independently of the noise, which triggers the emergence of Ph. 2;

[4] See Prop. D.11 for their definition.

[5] For example, $\ell_1 = \frac{2}{\pi}$, $\ell_2 = \frac{1}{\sqrt{2}}$, and $\ell_\infty = \sqrt{\frac{2}{\pi}}$.

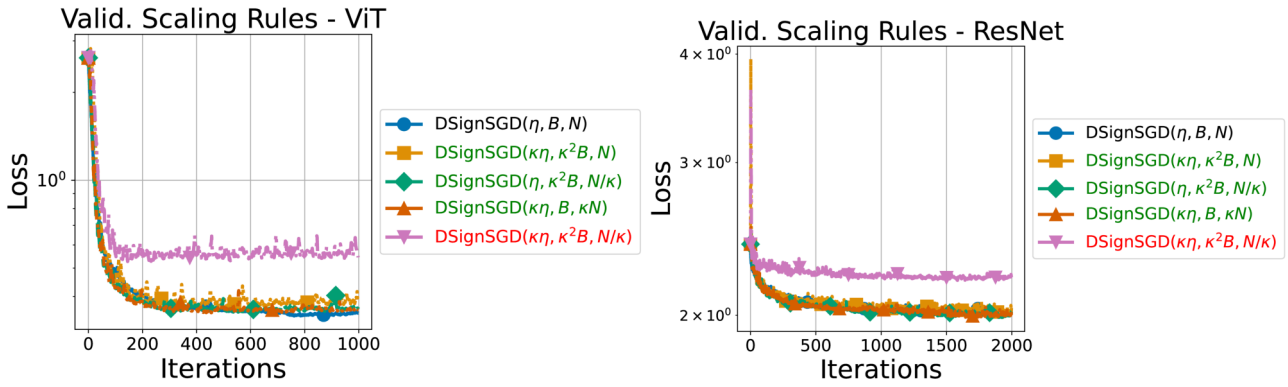

Figure 4: Validation of Scaling Rules: Consistently with Prop. D.15, DSignSGD run with hyperparameters that follow our scaling rule (marked in green in the legends) recover the performance of DSignSGD(η, B, N). The one that does not (marked in red) fails to do so. On the left, we plot the training loss of a ViT for *some* rules while on the right we do the same for a ResNet. Details in Appendix F.

2. During Ph. 2 and Ph. 3, the exponential decay of the loss strongly **depends on the noise distributional properties**: It scales inversely to the noise level $\sigma_{\mathcal{H},1}$ and proportionally to the fatness of the tails ℓ_ν, meaning that larger noise and fatter tails imply a slower convergence;

3. The asymptotic loss level **achieves a linear speedup with** N, scales inversely to ℓ_ν and proportionally to $\sigma_{\mathcal{H},1}$: More agents imply lower loss (see right of Fig. 3) while fatter tails and larger noise imply larger loss (Sec. 4 for a discussion and empirical validation).

The next theorem shows a general condition on the learning rate scheduler to ensure the convergence of DSignSGD for the general non-convex case. Interestingly, it sheds light on how DSignSGD reduces different gradient norms (L^1 and L^2) across different phases.

Theorem 3.13. *Let f be L-smooth, the learning rate scheduler η_t s.t. $\phi_t^i = \int_0^t (\eta_s)^i ds$, $\phi_t^1 \overset{t\to\infty}{\to} \infty$, $\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0$, $\Sigma_i \leq \sigma^2_{max,i}$, and $S_0 := f(X_0) - f(X_*)$. Then,*

1. *In **Phase 1**, $\|\nabla f(X_{\tilde{t}^1})\|_1 \leq \frac{S_0}{\phi_t^1} \overset{t\to\infty}{\to} 0$;*

2. *In **Phase 2**,*

$$\mathbb{E}\|\nabla f(X_{\tilde{t}^{(1,2)}})\|_2^2 + \frac{\hat{q}_\nu \sigma_{\mathcal{H},1}}{m_\nu \sqrt{B}} \mathbb{E}\|\nabla f(X_{\tilde{t}^{(2,2)}})\|_1 \quad (14)$$
$$\leq \frac{\sigma_{\mathcal{H},1}}{\phi_t^1 m_\nu \sqrt{B}} \left(S_0 + \frac{\eta L d \phi_t^2}{2N} \right) \overset{t\to\infty}{\to} 0;$$

3. *In **Phase 3**, $\mathbb{E}\|\nabla f(X_{\tilde{t}^3})\|_2^2$ is smaller than*

$$\frac{\sigma_{\mathcal{H},1}}{\phi_t^1 \ell_\nu \sqrt{B}} \left(S_0 + \frac{\eta L d \phi_t^2}{2N} \right) \overset{t\to\infty}{\to} 0. \quad (15)$$

Above, $\tilde{t}^1$, $\tilde{t}^{(1,2)}$, $\tilde{t}^{(2,2)}$, and $\tilde{t}^3$ are random times with distribution $\frac{\eta_t}{\phi_t^1}$.

Observations on Convergence - Non-convex:

1. DSignSGD **implicitly** minimizes the L^1 norm of the gradient in Ph. 1, a linear combination of L^1 and L^2 norm in Ph. 2, and transitions to optimizing the L^2 norm in Ph. 3;

2. Large and fat noise slow down the convergence;

3. Note that (95) derived convergence guarantees for a mixture of L^1 and L^2 norms. This mixture reduces to a rescaled L^1 norm when the gradients are large (similarly to our Ph. 1) and to a rescaled L^2 norm when the gradients are small (as in our Ph. 3). However, we highlight that our rates reveal *exactly* how all parameters affect the rate while in (95) these dependencies are *hidden* in the mixed norm.

We conclude by observing that *not all biased compressors* can handle fat noise, e.g. Top-k fails as well, while a slight modification is promising — See Fig. 10.

Scaling Rules: Preserving Performance

Under the simplifying assumption that $N \gg 1$, the following result provides intuitive scaling rules for DSignSGD, while Prop. D.15 presents the general cases. We validate some rules in Fig. 4 on a ViT and a ResNet. See Appendix G for the validation on a 124M GPT2.

Proposition 3.14. *Let the batch size be δB, learning rate $\kappa\eta$, αN agents, and $N \gg 1$. The scaling rule to preserve the performance indep. of δ, κ, α, is $\frac{\kappa}{\alpha\sqrt{\delta}} = 1$.*

Observations:

1. If $\alpha = 1$, this rule coincides with Adam's (21);

2. For example, while preserving the performance of DSignSGD run with η, B, and N, one can increase the batch size to $\kappa^2 B$, the learning rate to $\kappa\eta$, and

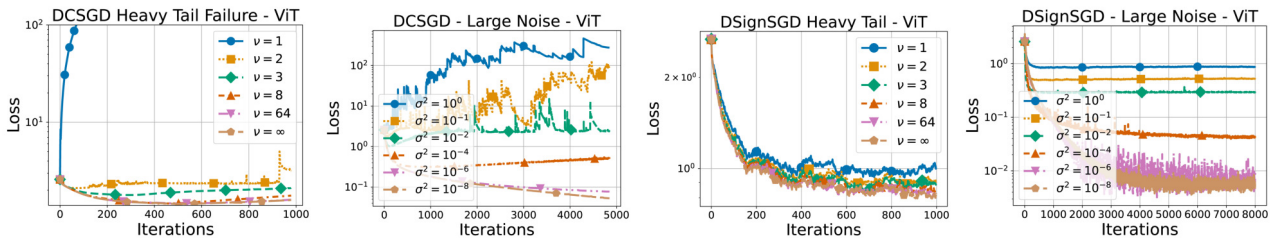

Figure 5: Empirical validation of the insights derived from Thm. 3.7 and Thm. 3.12: i) DCSGD cannot handle fat noise - The loss diverges if $\nu = 1$ and is non-stationary if $\nu = 2$ (**Left**); ii) The loss diverges more and more for larger noise (**Center-Left**); DSignSGD converges even when the noise is fat, although fatter noise implies less optimality (**Center-Right**); DSignSGD never diverges even when noise becomes increasingly larger (**Right**).

keep N agents. Alternatively, keep the learning rate to η but increase the batch size to $\kappa^2 B$ and decrease the number of agents down to $\frac{N}{\kappa}$.

Stationary Distribution. The stationary distribution of a process characterizes its behavior at convergence. Prop. D.16, shows that of DSignSGD: The main takeaway is that **the covariance matrix of the iterates scales linearly in the noise level**. In contrast, Prop. C.13 shows that the scaling is **quadratic** for DCSGD with k-Sparsification. These findings are novel and are empirically validated in Figure 9.

4 HEAVY AND LARGE NOISE

In this section, we recap our findings regarding the behavior of D(C)SGD and DSignSGD w.r.t. how fat, i.e. how heavy-tailed the noise is, and its level, i.e. its standard deviation or scale. We validate our results as we inject Gaussian or Student's t-distributed noise on the full gradient of a ViT trained on MNIST.

Theoretically and practically, **DCSGD diverges if the noise is fat**, i.e., does not admit bounded first or second moments (left of Figure 5). Provided that the noise admits a finite second moment, as per Thm. 3.7, DCSGD converges to an asymptotic loss level that scales **quadratically** with the noise level σ: The center-left of Figure 5 shows this on a ViT while Figure 11 validates the tightness of the bounds derived in Thm. 3.7 on a quadratic convex function for several noise levels.

In contrast, Theorem 3.12 shows that **DSignSGD converges even if the noise has an unbounded expected value**. In particular, the fatness of the tails influences both the convergence speed and the asymptotic loss level: **Fatter and larger** noise implies a **slower convergence** to a **larger** asymptotic level (center-right of Figure 5). Additionally, the asymptotic loss level of DSignSGD scales (approximately) **linearly** with the noise level: Figure 5 show this on a ViT while Figure 11 demonstrates the tightness of the bounds derived in Theorem 3.12 on a quadratic convex function for several noise levels.

5 CONCLUSIONS

We derived the first formal SDE models for DSGD, DCSGD, and DSignSGD, enabling us to elucidate the complex and different ways in which *unbiased* and *sign* compression interact with gradient noise. We started by showcasing the *tightness* of our analysis as we recovered and empirically validated the *best known* convergence results for DSGD and DCSGD: 1) We quantified how *unbiased* compression **slows down** the convergence of DCSGD w.r.t. DSGD, and showed that the noise level **does not impact** the convergence speed; 2) Unbiased compression and noise level interact nonlinearly by negatively affecting the asymptotic loss level of DCSGD w.r.t. DSGD. For DSignSGD, we 3) proved that *sign* compression implies that noise level **does influence** the speed of convergence as **larger noise slows it down**; 4) While the asymptotic loss level of DCSGD scales **quadratically** in the noise level, that of DSignSGD does so **linearly**; 5) DSignSGD is **resilient to heavy-tailed** noise and converges even when this has an unbounded expected value. Much differently, an **unbounded variance** of the noise is already enough for **DCSGD to diverge**. 6) Importantly, we prove that DSignSGD achieves **linear speedup**; 7) Finally, we derive **novel scaling rules** for DCSGD and DSignSGD, providing intuitive and actionable guidelines for selecting hyperparameters. These rules ensure that the performance of the algorithms is preserved, even allowing DCSGD to *recover* the performance of its *uncompressed* counterpart and DSignSGD to *preserve* it. Finally, we verify our results on a variety of deep learning architectures and datasets.

Future work. Our analysis can be extended to other practical optimizers, such as Top-k or DSignSGD with majority vote (13). Moreover, the insights derived from our SDE analysis provide a foundation for developing new optimization algorithms that integrate the strengths of current methods while addressing their limitations. Finally, it is possible to extend most of our results to the *heterogeneous* federated setting, up to some adjustments in the regularity of the local loss functions.

6 Acknowledgments

Enea Monzio Compagnoni, Rustem Islamov, and Aurelien Lucchi acknowledge the financial support of the Swiss National Foundation, SNF grant No 207392. Frank Norbert Proske acknowledges the financial support of the Norwegian Research Council (project No 274410) and MSCA4Ukraine (project No 101101923).

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. TensorFlow: a system for Large-Scale machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, 2016.

[2] Sungjin Ahn, Anoop Korattikara, and Max Welling. Bayesian posterior sampling via stochastic gradient Fisher scoring. *arXiv preprint arXiv:1206.6380*, 2012.

[3] Dan Alistarh, Christopher De Sa, and Nikola Konstantinov. The convergence of stochastic gradient descent in asynchronous shared memory. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, 2018.

[4] Jing An, Jianfeng Lu, and Lexing Ying. Stochastic modified equations for the asynchronous stochastic gradient descent. *Information and Inference: A Journal of the IMA*, 2020.

[5] Stefan Ankirchner and Stefan Perko. A comparison of continuous-time approximations to stochastic gradient descent. *Journal of Machine Learning Research*, 2024.

[6] Rotem Zamir Aviv, Ido Hakimi, Assaf Schuster, and Kfir Y Levy. Learning under delayed feedback: Implicitly adapting to gradient delays. *ICML*, 2021.

[7] Ghadir Ayache, Venkat Dassari, and Salim El Rouayheb. Walk for learning: A random walk approach for federated learning from heterogeneous data. *IEEE Journal on Selected Areas in Communications*, 2023.

[8] Imen Ayadi and Gabriel Turinici. Stochastic runge-kutta methods and adaptive sgd-g2 stochastic gradient descent. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021.

[9] Lukas Balles and Philipp Hennig. Dissecting Adam: The sign, magnitude and variance of stochastic gradients. In *International Conference on Machine Learning*, 2018.

[10] Martino Bardi and Hicham Kouhkouh. Deep relaxation of controlled stochastic gradient descent via singular perturbations. *arXiv preprint arXiv:2209.05564*, 2022.

[11] Aritz Bercher, Lukas Gonon, Arnulf Jentzen, and Diyora Salimova. Weak error analysis for stochastic gradient descent optimization algorithms. *arXiv preprint arXiv:2007.02723*, 2020.

[12] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. SignSGD: Compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[13] Jeremy Bernstein, Jiawei Zhao, Kamyar Azizzadenesheli, and Anima Anandkumar. signSGD with majority vote is communication efficient and fault tolerant. In *International Conference on Learning Representations*, 2019.

[14] Aleksandr Beznosikov, Samuel Horváth, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *Journal of Machine Learning Research*, 2023.

[15] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[16] Matias D. Cattaneo, Jason Matthew Klusowski, and Boris Shigida. On the implicit bias of adam, 2024.

[17] Peng Chen, Jianya Lu, and Lihu Xu. Approximation to stochastic variance reduced gradient langevin dynamics by stochastic delay differential equations. *Applied Mathematics & Optimization*, 2022.

[18] Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, pages 1683–1691. PMLR, 2014.

[19] Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, et al. Symbolic discovery of optimization algorithms. *Advances in neural information processing systems*, 2024.

[20] Enea Monzio Compagnoni, Luca Biggio, Antonio Orvieto, Frank Norbert Proske, Hans Kersting, and Aurelien Lucchi. An sde for modeling sam: Theory and insights. In *International Conference on Machine Learning*, pages 25209–25253. PMLR, 2023.

[21] Enea Monzio Compagnoni, Tianlin Liu, Rustem Islamov, Frank Norbert Proske, Antonio Orvieto, and Aurelien Lucchi. Adaptive methods through the lens of SDEs: Theoretical insights on the role of noise. In *The Thirteenth International Conference on Learning Representations*, 2025.

[22] Enea Monzio Compagnoni, Antonio Orvieto, Hans Kersting, Frank Proske, and Aurelien Lucchi. Sdes for minimax optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 4834–4842. PMLR, 2024.

[23] Laurent Condat, Artavazd Maranjyan, and Peter Richtárik. LoCoDL: Communication-efficient distributed learning with local training and compression. *arXiv preprint arXiv:2403.04348*, 2024.

[24] Laurent Condat, Kai Yi, and Peter Richtárik. EF-BV: A unified theory of error feedback and variance reduction mechanisms for biased and unbiased compression in distributed optimization. *Advances in Neural Information Processing Systems*, 2022.

[25] Zhuo-Xu Cui, Qibin Fan, and Cui Jia. Momentum methods for stochastic optimization over time-varying directed networks. *Signal Processing*, 2020.

[26] Marc Dambrine, Ch Dossal, Bénédicte Puig, and Aude Rondepierre. Stochastic differential equations for modeling first order optimization methods. *SIAM Journal on Optimization*, 2024.

[27] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Marc'aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, et al. Large scale distributed deep networks. *Advances in neural information processing systems*, 2012.

[28] Pierre Del Moral and Angele Niclas. A taylor expansion of the square root matrix function. *Journal of Mathematical Analysis and Applications*, 465(1):259–266, 2018.

[29] Li Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 2012.

[30] Yuyang Deng, Mohammad Mahdi Kamani, Pouria Mahdavinia, and Mehrdad Mahdavi. Distributed personalized empirical risk minimization. *Advances in Neural Information Processing Systems*, 2024.

[31] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[32] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at Scale. In *International Conference on Learning Representations*, 2021.

[33] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[34] Ilyas Fatkhullin, Alexander Tyurin, and Peter Richtárik. Momentum provably improves error feedback! *Advances in Neural Information Processing Systems*, 2024.

[35] Xavier Fontaine, Valentin De Bortoli, and Alain Durmus. Convergence rates and approximation results for SGD and its continuous-time counterpart. In *Conference on Learning Theory*, 2021.

[36] Yuan Gao, Rustem Islamov, and Sebastian Stich. EControl: Fast distributed optimization with compression and error control. *arXiv preprint arXiv:2311.05645*, 2023.

[37] Guillaume Garrigos and Robert M Gower. Handbook of convergence theorems for (stochastic) gradient methods. *arXiv preprint arXiv:2301.11235*, 2023.

[38] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, 2015.

[39] Benjamin Gess, Sebastian Kassing, and Vitalii Konarovskyi. Stochastic modified flows, mean-field limits and dynamics of stochastic gradient descent. *Journal of Machine Learning Research*, 2024.

[40] Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. A unified theory of SGD: Variance reduction, sampling, quantization and coordinate descent. In *International Conference on Artificial Intelligence and Statistics*, 2020.

[41] Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, 2021.

[42] Eduard Gorbunov, Abdurakhmon Sadiev, Marina Danilova, Samuel Horváth, Gauthier Gidel, Pavel Dvurechensky, Alexander Gasnikov, and Peter Richtárik. High-probability convergence for composite and distributed stochastic minimization and variational inequalities with heavy-tailed noise. *arXiv preprint arXiv:2310.01860*, 2023.

[43] Haotian Gu, Xin Guo, and Xinyu Li. Adversarial training for gradient descent: Analysis through its continuous-time approximation. *arXiv preprint arXiv:2105.08037*, 2021.

[44] Mert Gurbuzbalaban, Umut Simsekli, and Lingjiong Zhu. The heavy-tail phenomenon in SGD. In *International Conference on Machine Learning*, 2021.

[45] Moritz Hardt, Tengyu Ma, and Benjamin Recht. Gradient descent learns linear dynamical systems. *Journal of Machine Learning Research*, 19(29):1–44, 2018.

[46] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 2020.

[47] Samuel Horváth, Chen-Yu Ho, Ludovit Horvath, Atal Narayan Sahu, Marco Canini, and Peter Richtárik. Natural compression for distributed deep learning. In *Mathematical and Scientific Machine Learning*, 2022.

[48] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Peter Richtárik, and Sebastian Stich. Stochastic distributed learning with gradient quantization and double-variance reduction. *Optimization Methods and Software*, 2023.

[49] Wenqing Hu, Chris Junchi Li, and Xiang Zhou. On the global convergence of continuous–time stochastic heavy–ball method for nonconvex optimization. In *2019 IEEE International Conference on Big Data (Big Data)*, 2019.

[50] Nobuyuki Ikeda and Shinzo Watanabe. *Stochastic differential equations and diffusion processes*. Elsevier, 2014.

[51] Rustem Islamov, Niccolò Ajroldi, Antonio Orvieto, and Aurelien Lucchi. Loss landscape characterization of neural networks without over-parametrization. *arXiv preprint arXiv:2410.12455*, 2024.

[52] Rustem Islamov, Xun Qian, Slavomír Hanzely, Mher Safaryan, and Peter Richtárik. Distributed newton-type methods with communication compression and bernoulli aggregation. *Transactions on Machine Learning Research*, 2023.

[53] Rustem Islamov, Xun Qian, and Peter Richtárik. Distributed second order methods with fast rates and compressed communication. In *International conference on machine learning*, pages 4617–4628. PMLR, 2021.

[54] Rustem Islamov, Mher Safaryan, and Dan Alistarh. AsGrad: A sharp unified analysis of asynchronous-SGD algorithms. In *International Conference on Artificial Intelligence and Statistics*, 2024.

[55] Stanisław Jastrzebski, Zachary Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. Three factors influencing minima in sgd. *ICANN 2018*, 2018.

[56] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, 2017.

[57] Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture*, 2017.

[58] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pages 795–811. Springer, 2016.

[59] Sarit Khirirat, Hamid Reza Feyzmahdavian, and Mikael Johansson. Distributed learning with compressed gradients. *arXiv preprint arXiv:1806.06573*, 2018.

[60] Sarit Khirirat, Eduard Gorbunov, Samuel Horváth, Rustem Islamov, Fakhri Karray, and Peter Richtárik. Clip21: Error feedback for gradient clipping. *arXiv preprint: arXiv 2305.18929*, 2023.

[61] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, 2020.

[62] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Toronto, ON, Canada*, 2009.

[63] Daniel Kunin, Javier Sagastuy-Brena, Lauren Gillespie, Eshed Margalit, Hidenori Tanaka, Surya Ganguli, and Daniel LK Yamins. The limiting dynamics of SGD: Modified loss, phase-space oscillations, and anomalous diffusion. *Neural Computation*, 2023.

[64] Frederik Kunstner, Robin Yadav, Alan Milligan, Mark Schmidt, and Alberto Bietti. Heavy-tailed class imbalance and why adam outperforms gradient descent on language models. *arXiv preprint arXiv:2402.19449*, 2024.

[65] Alberto Lanconelli and Christopher SA Lauria. A note on diffusion limits for stochastic gradient descent. *arXiv preprint arXiv:2210.11257*, 2022.

[66] Kfir Y Levy. The power of normalization: Faster evasion of saddle points. *arXiv preprint arXiv:1611.04831*, 2016.

[67] Boyue Li and Yuejie Chi. Convergence and privacy of decentralized nonconvex optimization with gradient clipping and communication compression. *arXiv preprint arXiv:2305.09896*, 2023.

[68] Lei Li and Yuliang Wang. On uniform-in-time diffusion approximation for stochastic gradient descent. *arXiv preprint arXiv:2207.04922*, 2022.

[69] Qianxiao Li, Cheng Tai, and E Weinan. Stochastic modified equations and adaptive stochastic gradient algorithms. In *International Conference on Machine Learning*, pages 2101–2110. PMLR, 2017.

[70] Qianxiao Li, Cheng Tai, and E Weinan. Stochastic modified equations and dynamics of stochastic gradient algorithms i: Mathematical foundations. *The Journal of Machine Learning Research*, 2019.

[71] Zhiyuan Li, Sadhika Malladi, and Sanjeev Arora. On the validity of modeling SGD with stochastic differential equations (SDEs). In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[72] Zhiyuan Li, Yi Wang, and Zhiren Wang. Fast equilibrium of SGD in generic situations. In *The Twelfth International Conference on Learning Representations*, 2023.

[73] Zhize Li and Peter Richtárik. A unified analysis of stochastic gradient methods for nonconvex federated optimization. *arXiv preprint arXiv:2006.07013*, 2020.

[74] Chaoyue Liu, Dmitriy Drusvyatskiy, Misha Belkin, Damek Davis, and Yian Ma. Aiming towards the minimizers: fast convergence of sgd for overparametrized problems. *Advances in neural information processing systems*, 36, 2024.

[75] Tianyi Liu, Zhehui Chen, Enlu Zhou, and Tuo Zhao. A diffusion approximation theory of momentum stochastic gradient descent in nonconvex optimization. *Stochastic Systems*, 2021.

[76] Sadhika Malladi, Kaifeng Lyu, Abhishek Panigrahi, and Sanjeev Arora. On the SDEs and scaling rules for adaptive gradient algorithms. In *Advances in Neural Information Processing Systems*, 2022.

[77] Stephan Mandt, Matthew Hoffman, and David Blei. A variational analysis of stochastic gradient algorithms. In *International conference on machine learning*, 2016.

[78] Stephan Mandt, Matthew D Hoffman, and David M Blei. Stochastic gradient descent as approximate bayesian inference. *JMLR*, 2017.

[79] Xuerong Mao. *Stochastic differential equations and applications*. Elsevier, 2007.

[80] Othmane Marfoq, Giovanni Neglia, Laetitia Kameni, and Richard Vidal. Federated learning for data streams. In *International Conference on Artificial Intelligence and Statistics*, 2023.

[81] Rodrigo Maulen-Soto, Jalal Fadili, Hedy Attouch, and Peter Ochs. Stochastic inertial dynamics via time scaling and averaging. *arXiv preprint arXiv:2403.16775*, 2024.

[82] Rodrigo Ignacio Maulén Soto. A continuous-time model of stochastic gradient descent: convergence rates and complexities under lojasiewicz inequality. *Universidad de Chile*, 2021.

[83] GN Mil'shtein. Weak approximation of solutions of systems of stochastic differential equations. *Theory of Probability & Its Applications*, 1986.

[84] Konstantin Mishchenko, Francis Bach, Mathieu Even, and Blake E Woodworth. Asynchronous SGD beats minibatch SGD under arbitrary delays. *Advances in Neural Information Processing Systems*, 2022.

[85] Konstantin Mishchenko, Eduard Gorbunov, Martin Takáč, and Peter Richtárik. Distributed learning with compressed gradient differences. *Optimization Methods and Software*, 2024.

[86] Bernt Øksendal. When is a stochastic integral a time change of a diffusion? *Journal of theoretical probability*, 1990.

[87] Antonio Orvieto and Aurelien Lucchi. Continuous-time models for stochastic optimization algorithms. *Advances in Neural Information Processing Systems*, 2019.

[88] Courtney Paquette, Kiwon Lee, Fabian Pedregosa, and Elliot Paquette. SGD in the large: Average-case analysis, asymptotics, and stepsize criticality. In *Conference on Learning Theory*. PMLR, 2021.

[89] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011.

[90] Constantin Philippenko and Aymeric Dieuleveut. Compressed and distributed least-squares regression: convergence rates with applications to federated learning. *Journal of Machine Learning Research*, 25(288):1–80, 2024.

[91] Tomaso Poggio, Kenji Kawaguchi, Qianli Liao, Brando Miranda, Lorenzo Rosasco, Xavier Boix, Jack Hidary, and Hrushikesh Mhaskar. Theory of deep learning III: explaining the non-overfitting puzzle. *arXiv preprint arXiv:1801.00173*, 2017.

[92] Xun Qian, Rustem Islamov, Mher Safaryan, and Peter Richtárik. Basis matters: better communication-efficient second order methods for federated learning. *arXiv preprint arXiv:2111.01847*, 2021.

[93] Peter Richtárik, Igor Sokolov, and Ilyas Fatkhullin. EF21: A new, simpler, theoretically better, and practically faster error feedback. *Advances in Neural Information Processing Systems*, 2021.

[94] Mher Safaryan, Rustem Islamov, Xun Qian, and Peter Richtárik. FedNL: Making newton-type methods applicable to federated learning. *arXiv preprint arXiv:2106.02969*, 2021.

[95] Mher Safaryan and Peter Richtarik. Stochastic sign descent methods: New algorithms and better theory. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[96] Amedeo Sapio, Marco Canini, Chen-Yu Ho, Jacob Nelson, Panos Kalnis, Changhoon Kim, Arvind Krishnamurthy, Masoud Moshref, Dan Ports, and Peter Richtárik. Scaling distributed machine learning with {In-Network} aggregation. In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*, 2021.

[97] Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. In *Interspeech*, 2014.

[98] Ohad Shamir and Nathan Srebro. Distributed stochastic optimization and learning. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2014.

[99] Umut Simsekli, Levent Sagun, and Mert Gurbuzbalaban. A tail-index analysis of stochastic gradient noise in deep neural networks. In *International Conference on Machine Learning*, 2019.

[100] Samuel L. Smith, Benoit Dherin, David G. T. Barrett, and Soham De. On the origin of implicit regularization in stochastic gradient descent. *arXiv preprint arXiv: 2101.12176*, 2021.

[101] Rodrigo Maulen Soto, Jalal Fadili, and Hedy Attouch. An SDE perspective on stochastic convex optimization. *arXiv preprint arXiv:2207.02750*, 2022.

[102] Mandt Stephan, Matthew D Hoffman, David M Blei, et al. Stochastic gradient descent as approximate bayesian inference. *Journal of Machine Learning Research*, 2017.

[103] Liqun Su and Vincent KN Lau. Accelerated federated learning over wireless fading channels with adaptive stochastic momentum. *IEEE Internet of Things Journal*, 2023.

[104] Chao Sun. Distributed stochastic optimization under heavy-tailed noises. *arXiv preprint arXiv:2312.15847*, 2023.

[105] Jianhui Sun, Ying Yang, Guangxu Xun, and Aidong Zhang. Scheduling hyperparameters to improve generalization: From centralized SGD to asynchronous SGD. *ACM Transactions on Knowledge Discovery from Data*, 2023.

[106] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[107] Thijs Vogels, Sai Praneeth Karimireddy, and Martin Jaggi. PowerSGD: Practical low-rank gradient compression for distributed optimization. *Advances in Neural Information Processing Systems*, 2019.

[108] Jialei Wang, Weiran Wang, and Nathan Srebro. Memory and communication efficient distributed stochastic optimization with minibatch prox. In *Conference on Learning Theory*, 2017.

[109] Yazhen Wang and Shang Wu. Asymptotic analysis via stochastic differential equations of gradient descent algorithms in statistical and computational paradigms. *Journal of machine learning research*, 2020.

[110] Ziqiao Wang and Yongyi Mao. Two facets of SDE under an information-theoretic lens: Generalization of SGD via training trajectories and via terminal states. *arXiv preprint arXiv:2211.10691*, 2022.

[111] Henry Wolkowicz and George PH Styan. Bounds for eigenvalues using traces. *Linear algebra and its applications*, 29:471–506, 1980.

[112] Jingfeng Wu, Wenqing Hu, Haoyi Xiong, Jun Huan, Vladimir Braverman, and Zhanxing Zhu. On the noisy gradient descent that generalizes as SGD. In *International Conference on Machine Learning*, 2020.

[113] Ke Liang Xiao, Noah Marshall, Atish Agarwala, and Elliot Paquette. Exact risk curves of signsgd in high-dimensions: Quantifying preconditioning and noise-compression effects. *arXiv preprint arXiv:2411.12135*, 2024.

[114] Zeke Xie, Li Yuan, Zhanxing Zhu, and Masashi Sugiyama. Positive-negative momentum: Manipulating stochastic gradient noise to improve generalization. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

[115] Haibo Yang, Peiwen Qiu, and Jia Liu. Taming fat-tailed ("heavier-tailed" with potentially infinite variance) noise in federated learning. *Advances in Neural Information Processing Systems*, 2022.

[116] Shuhua Yu, Dusan Jakovetic, and Soummya Kar. Smoothed gradient clipping and error feedback for distributed optimization under heavy-tailed noise. *arXiv preprint arXiv:2310.16920*, 2023.

[117] Yue Yu, Jiaxiang Wu, and Longbo Huang. Double quantization for communication-efficient distributed optimization. *Advances in neural information processing systems*, 2019.

[118] Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models? *Advances in Neural Information Processing Systems*, 2020.

[119] Zhongwang Zhang, Yuqing Li, Tao Luo, and Zhi-Qin John Xu. Stochastic modified equations and dynamics of dropout algorithm. *arXiv preprint arXiv:2305.15850*, 2023.

[120] Jim Zhao, Aurelien Lucchi, Frank Norbert Proske, Antonio Orvieto, and Hans Kersting. Batch size selection by stochastic optimal control. In *Has it Trained Yet? NeurIPS 2022 Workshop*, 2022.

[121] Shenyi Zhao, Gong-Duo Zhang, Ming-Wei Li, and Wu-Jun Li. Proximal SCOPE for distributed sparse learning. *Advances in Neural Information Processing Systems*, 2018.

[122] Xiang Zhou, Huizhuo Yuan, Chris Junchi Li, and Qingyun Sun. Stochastic modified equations for continuous limit of stochastic admm. *arXiv preprint arXiv:2003.03532*, 2020.

[123] Yuhua Zhu and Lexing Ying. A sharp convergence rate for a model equation of the asynchronous stochastic gradient descent. *Communications in Mathematical Sciences*, 2021.

[124] Zhanxing Zhu, Jingfeng Wu, Bing Yu, Lei Wu, and Jinwen Ma. The anisotropic noise in stochastic gradient descent: Its behavior of escaping from sharp minima and regularization effects. *ICML*, 2019.

Checklist

1. For all models and algorithms presented, check if you include:

(a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]

(b) An analysis of the properties and complexity (time, space, sample size) of any algoriThm. [Yes]

(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Not Applicable]

2. For any theoretical claim, check if you include:

(a) Statements of the full set of assumptions of all theoretical results. [Yes]

(b) Complete proofs of all theoretical results. [Yes]

(c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:

(a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [No]

(b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

(c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

(a) Citations of the creator If your work uses existing assets. [Yes]

(b) The license information of the assets, if applicable. [Yes]

(c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

(d) Information about consent from data providers/curators. [Not Applicable]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

APPENDIX

A Stochastic calculus

In this section, we summarize some important results in the analysis of Stochastic Differential Equations (79, 86). The notation and the results in this section will be used extensively in all proofs in this paper. We assume the reader to have some familiarity with Brownian motion and with the definition of stochastic integral (Ch. 1.4 and 1.5 in (79)).

A.1 Itô's Lemma

We start with some notation: Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, \mathbb{P})$ be a filtered probability space. We say that an event $E \in \mathcal{F}$ holds almost surely (a.s.) in this space if $\mathbb{P}(E) = 1$. We call $\mathcal{L}^p([a,b], \mathbb{R}^d)$, with $p > 0$, the family of $\mathbb{R}^d$-valued $\mathcal{F}_t$-adapted processes $\{f_t\}_{a\leq t\leq b}$ such that

$$\int_a^b \|f_t\|^p dt \leq \infty.$$

Moreover, we denote by $\mathcal{M}^p([a,b], \mathbb{R}^d)$, with $p > 0$, the family of $\mathbb{R}^d$-valued processes $\{f_t\}_{a\leq t\leq b}$ in $\mathcal{L}([a,b], \mathbb{R}^d)$ such that $\mathbb{E}\left[\int_a^b \|f_t\|^p dt\right] \leq \infty$. We will write $h \in \mathcal{L}^p\left(\mathbb{R}_+, \mathbb{R}^d\right)$, with $p > 0$, if $h \in \mathcal{L}^p\left([0,T], \mathbb{R}^d\right)$ for every $T > 0$. Similar definitions hold for matrix-valued functions using the Frobenius norm $\|A\| := \sqrt{\sum_{ij} |A_{ij}|^2}$.

Let $W = \{W_t\}_{t\geq 0}$ be a one-dimensional Brownian motion defined on our probability space and let $X = \{X_t\}_{t\geq 0}$ be an $\mathcal{F}_t$-adapted process taking values on $\mathbb{R}^d$.

Definition A.1. Let the *drift* be $b \in \mathcal{L}^1\left(\mathbb{R}_+, \mathbb{R}^d\right)$ and the diffusion term be $\sigma \in \mathcal{L}^2\left(\mathbb{R}_+, \mathbb{R}^{d\times m}\right)$. X_t is an Itô process if it takes the form

$$X_t = x_0 + \int_0^t b_s ds + \int_0^t \sigma_s dW_s.$$

We shall say that X_t has the stochastic differential

$$dX_t = b_t dt + \sigma_t dW_t. \tag{16}$$

Theorem A.2 (Itô's Lemma)**.** *Let X_t be an Itô process with stochastic differential $dX_t = b_t dt + \sigma_t dW_t$. Let $f(x,t)$ be twice continuously differentiable in x and continuously differentiable in t, taking values in $\mathbb{R}$. Then $f(X_t, t)$ is again an Itô process with stochastic differential*

$$df(X_t,t) = \partial_t f(X_t,t))dt + \langle \nabla f(X_t,t), b_t\rangle dt + \frac{1}{2} Tr\left(\sigma_t \sigma_t^\top \nabla^2 f(X_t,t)\right) dt + \langle \nabla f(X_t,t), \sigma_t\rangle dW_t. \tag{17}$$

A.2 Stochastic Differential Equations

Stochastic Differential Equations (SDEs) are equations of the form

$$dX_t = b(X_t,t)dt + \sigma(X_t,t)dW_t.$$

First of all, we need to define what it means for a stochastic process $X = \{X_t\}_{t\geq 0}$ with values in $\mathbb{R}^d$ to solve an SDE.

Definition A.3. Let X_t be as above with deterministic initial condition $X_0 = x_0$. Assume $b : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ and $\sigma : \mathbb{R}^d \times [0,T] \to \mathbb{R}^{d\times m}$ are Borel measurable; X_t is called a solution to the corresponding SDE if

1. X_t is continuous and $\mathcal{F}_t$-adapted;
2. $b \in \mathcal{L}^1\left([0,T], \mathbb{R}^d\right)$;

3. $\sigma \in \mathcal{L}^2\left([0,T], \mathbb{R}^{d\times m}\right)$;

4. For every $t \in [0,T]$

$$X_t = x_0 + \int_0^t b(X_s, s)ds + \int_0^t \sigma(X_s, s)dW(s) \quad a.s.$$

Moreover, the solution X_t is said to be unique if any other solution $X_t^\star$ is such that

$$\mathbb{P}\left\{X_t = X_t^\star, \text{ for all } 0 \le t \le T\right\} = 1.$$

Notice that since the solution to an SDE is an Itô process, we can use Itô's Lemma. The following theorem gives a sufficient condition on b and σ for the existence of a solution to the corresponding SDE.

Theorem A.4. *Assume that there exist two positive constants $\bar{K}$ and K such that*

1. *(Global Lipschitz condition) for all $x, y \in \mathbb{R}^d$ and $t \in [0,T]$*

$$\max\{\|b(x,t) - b(y,t)\|^2,\ \|\sigma(x,t) - \sigma(y,t)\|^2\} \le \bar{K}\|x - y\|^2;$$

2. *(Linear growth condition) for all $x \in \mathbb{R}^d$ and $t \in [0,T]$*

$$\max\{\|b(x,t)\|^2,\ \|\sigma(x,t)\|^2\} \le K(1 + \|x\|^2).$$

Then, there exists a unique solution X_t to the corresponding SDE, and $X_t \in \mathcal{M}^2([0,T], \mathbb{R}^d)$.

Numerical approximation. Often, SDEs are solved numerically. The simplest algorithm to provide a sample path $(\hat{x}_k)_{k\ge 0}$ for X_t, so that $X_{k\Delta t} \approxeq \hat{x}_k$ for some small Δt and for all $k\Delta t \le M$ is called Euler-Maruyama (Algorithm 1). For more details on this integration method and its approximation properties, the reader can check (79).

Algorithm 1 Euler-Maruyama Integration Method for SDEs

input The drift b, the volatility σ, and the initial condition x_0.
  Fix a stepsize Δt;
  Initialize $\hat{x}_0 = x_0$;
  $k = 0$;
  while $k \le \lfloor \frac{T}{\Delta t} \rfloor$ **do**
    Sample some d-dimensional Gaussian noise $Z_k \sim \mathcal{N}(0, I_d)$;
    Compute $\hat{x}_{k+1} = \hat{x}_k + \Delta t\ b(\hat{x}_k, k\Delta t) + \sqrt{\Delta t}\ \sigma(\hat{x}_k, k\Delta t)Z_k$;
    $k = k + 1$;
  end while
output The approximated sample path $(\hat{x}_k)_{0\le k\le \lfloor \frac{T}{\Delta t} \rfloor}$.

B Theoretical framework - Weak Approximation

In this section, we introduce the theoretical framework used in the paper, together with its assumptions and notations.

First of all, many proofs will use Taylor expansions in powers of η. For ease of notation, we introduce the shorthand that whenever we write $\mathcal{O}(\eta^\alpha)$, we mean that there exists a function $K(x) \in G$ such that the error terms are bounded by $K(x)\eta^\alpha$. For example, we write

$$b(x + \eta) = b_0(x) + \eta b_1(x) + \mathcal{O}\left(\eta^2\right)$$

to mean: there exists $K \in G$ such that

$$|b(x + \eta) - b_0(x) - \eta b_1(x)| \le K(x)\eta^2.$$

Additionally, we introduce the following shorthand:

- A multi-index is $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that $\alpha_j \in \{0, 1, 2, \ldots\}$;
- $|\alpha| := \alpha_1 + \alpha_2 + \cdots + \alpha_n$;
- $\alpha! := \alpha_1!\alpha_2! \cdots \alpha_n!$;
- For $x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$, we define $x^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$;
- For a multi-index β, $\partial_\beta^{|\beta|} f(x) := \frac{\partial^{|\beta|}}{\partial_{x_1}^{\beta_1} \partial_{x_2}^{\beta_2} \ldots \partial_{x_n}^{\beta_n}} f(x)$;
- We also denote the partial derivative with respect to x_i by ∂_{e_i}.

Definition B.1 (G Set)**.** Let G denote the set of continuous functions $\mathbb{R}^d \to \mathbb{R}$ of at most polynomial growth, i.e. $g \in G$ if there exists positive integers $\nu_1, \nu_2 > 0$ such that $|g(x)| \leq \nu_1 \left(1 + |x|^{2\nu_2}\right)$, for all $x \in \mathbb{R}^d$.

Definition B.2 ($\mathcal{C}_b^k(\mathbb{R}^n, \mathbb{R})$)**.** $\mathcal{C}_b^k(\mathbb{R}^n, \mathbb{R})$ denotes the space of functions whose k-th derivatives are bounded.

B.1 Assumptions.

In general, we assume some regularity in the loss function.

Assumption B.3. Assume that the following conditions on $f, f_i \in \mathcal{C}_b^8(\mathbb{R}^n, \mathbb{R})$, and their gradients are satisfied:

- $\nabla f, \nabla f_i$ satisfy a Lipschitz condition: there exists $L > 0$ such that

$$|\nabla f(u) - \nabla f(v)| + \sum_{i=1}^{n} |\nabla f_i(u) - \nabla f_i(v)| \leq L|u - v|;$$

- f, f_i and its partial derivatives up to order 7 belong to G;
- $\nabla f, \nabla f_i$ satisfy a growth condition: there exists $M > 0$ such that

$$|\nabla f(x)| + \sum_{i=1}^{n} |\nabla f_i(x)| \leq M(1 + |x|).$$

Regarding the gradient noise, each optimizer has its mild assumptions which are weaker or in line with the literature.

DSGD

1. The covariance matrices $\Sigma_i(x)$ are Definite Positive;
2. Their square roots $\sqrt{\Sigma_i}(x)$ are: In G together with their derivatives, Lipschitz, bounded, and satisfy Affine Growth (76).

DCSGD Additionally w.r.t. DSGD, DCSGD requires:

1. The gradient noise $Z(x)$ admits a strictly positive density function g_x for all x and require that $g : \mathbb{R}^n \times \mathbb{R}^n \to [0, \infty)$ s.t. $(x, y) \mapsto g_x(y)$ is in $C^8(\mathbb{R}^n \times \mathbb{R}^n)$ such that all partial derivatives of g up to order 8 are integrable with respect to y and s.t. their L^1-norms are uniformly bounded in x. This assumption covers Gaussian and Student's t, thus being *more general than the literature*. Indeed, the Gaussianity of the noise is commonly assumed: Among others, see (2, 18, 77, 102, 124, 112, 114), while (55) offers an intuitive justification as well;

2. Bounded and closed domain (98, 108, 121, 117, 6, 7, 80, 30): This assumption is not restrictive in our case. Indeed, our contribution regarding DCSGD is not to prove their convergence, which has been proven before (59, 73), but rather the scaling rules in Prop. 3.9. Since convergence has already been guaranteed, we can assume the domain to be closed and bounded without loss of generality while still providing insightful and actionable results. Additionally, this is also assumed in the seminal paper for this theoretical framework (70);

3. For all compact sets K
$$\sup_{x\in K} |g(x,\cdot)| \in L^1(\mathbb{R}^n),$$
which of course covers the Gaussian case, *thus being more general than the literature.*

DSignSGD On top of the assumptions 1. and 3. of DCSGD, we need the functions in Eq. 19 to be in G, which, as we show below, covers Gaussian and Student's t, *thus being more general than the literature.*

Remark All the assumptions above are *in line with or more general than those commonly found in the literature.* In line with *Remark 11* of the seminal paper (70), we observe that while some of these assumptions might seem strong, loss functions in applications have inward pointing gradients for sufficiently large x. Therefore, we could simply modify the loss to satisfy the assumptions above.

Regarding the drift and diffusion coefficients, we highlight that many papers in the literature following this framework do not check for their regularity before applying the approximation theorems (49, 4, 123, 25, 82, 110, 20, 22, 69). At first sight, it would seem that not even the seminal paper (70) checks these conditions carefully. However, a deeper investigation shows that they are restricting their analysis to compact sets to leverage the regularity and convergence properties of mollifiers: The assumption regarding the compactness of the domain is not highlighted nor assumed in any part of the paper. Therefore, we conclude that, willingly or not, most papers are implicitly making these assumptions.

B.2 Technical Results

In this subsection, we provide some results that will be instrumental in the derivation of the SDEs.

Lemma B.4. *Assume the existence of a probability density g_x of the gradient noise $Z(x)$ for all x and require that $g : \mathbb{R}^n \times \mathbb{R}^n \to [0,\infty)\,;\,(x,y) \mapsto g_x(y)$ is in $C^8(\mathbb{R}^n \times \mathbb{R}^n)$ such that all partial derivatives of g up to order 8 are integrable with respect to y and such that their L^1-norms are uniformly bounded in x. Further, let $f \in C^8(\mathbb{R}^n)$ and $h : \mathbb{R}^n \to \mathbb{R}$ be a bounded Borel measurable function. Define the function k by*
$$k(x) = \mathbb{E}\left[h(\nabla f_\gamma(x))\right].$$
Then there exists a version $\widehat{k}$ of k with $\widehat{k} \in C_b^7(\mathbb{R}^n)$.

Proof. Let φ be smooth and compactly supported. Then for all multiindices β with $|\beta| \le 8$, substitution, Fubini's theorem, and integration by parts imply that
$$\begin{aligned}\int_{\mathbb{R}^n} k(x)\partial_\beta^{|\beta|}\varphi(x)dx &= \int_{\mathbb{R}^n} \mathbb{E}\left[h(\nabla f_\gamma(x))\right]\partial_\beta^{|\beta|}\varphi(x)dx \\ &= \int_{\mathbb{R}^n}\int_{\mathbb{R}^n} h(y)g_x(y-\nabla f(x))dy\partial_\beta^{|\beta|}\varphi(x)dx \\ &= (-1)^{|\beta|}\int_{\mathbb{R}^n}\int_{\mathbb{R}^n} h(y)\partial_\beta^{|\beta|}(g_x(y-\nabla f(x)))dy\varphi(x)dx.\end{aligned}$$
So
$$\int_{\mathbb{R}^n} h(y)\partial_\beta^{|\beta|}(g_x(y-\nabla f(x)))dy$$
is a weak derivative $\partial_\beta^{|\beta|}k$ of k on any bounded open set. For compact sets K we obtain that
$$\begin{aligned}&\int_K \left|\int_{\mathbb{R}^n} h(y)\partial_\beta^{|\beta|}(g_x(y-\nabla f(x)))dy\right|^p dx \\ \le\ & \|h\|_\infty^p\,\lambda^n(K)\left(\sup_{x\in\mathbb{R}^n}\int_{\mathbb{R}^n}\left|\partial_\beta^{|\beta|}(g_x(y-\nabla f(x)))\right|dy\right)^p < \infty\end{aligned}$$

for all $p \geq 2$ because of our assumptions on g and f and substitution (λ^n Lebesgue measure). So it follows from Sobolev embeddings with respect to Hölder spaces that for all bounded and open sets Ω there exists a version $\widehat{k}$ of k such that $\widehat{k} \in C^7(\Omega)$. The latter version can be extended to $\Omega = \mathbb{R}^n$, which we also denote by $\widehat{k}$. Since $\partial_\beta^{|\beta|} k$ is bounded for $|\beta| \leq 8$, we conclude that $\widehat{k} \in C_b^7(\mathbb{R}^n)$. □

Lemma B.5. *Assuming that for all compact sets K*

$$\sup_{x \in K} |g(x, \cdot)| \in L^1(\mathbb{R}^n),$$

and the positivity of the density functions, we have that for $m = 1, \ldots, 7$ that

$$\left\| \partial_{j_1} \ldots \partial_{j_m} A^{1/2}(x) \right\| \leq C l_m(x), \tag{18}$$

where the function $l_m(x)$ is defined as

$$\begin{aligned} l_m(x) &:= \sum_{r=0}^{m-1} \left(\frac{1}{m(x) + s(x)(n-1)^{1/2}} \left(1 + \frac{2s(x)(n-1)^{1/2}}{m(x) - s(x)(n-1)^{-1/2}} \right) \right)^{-(r+1/2)} \\ &\quad \times \max_{|\beta| \leq m} \left\| \partial_\beta^{|\beta|} A(x) \right\|^{r+1}. \end{aligned} \tag{19}$$

Proof. To prove this, we need the fact that the Fréchet derivatives of the square root function φ can be represented as follows (see Theorem 1.1 in (28)):

$$\nabla \varphi(A)[H] = \int_0^\infty e^{-t\varphi(A)} H e^{-t\varphi(A)} dt,$$

and higher derivatives of order $m \geq 2$ are given by

$$\begin{aligned} \nabla^m \varphi(A)[H, \ldots, H] &= -\nabla \varphi(A) \left[\sum_{p+q=m-2} \frac{m!}{(p+1)!(q+1)!} (\nabla^{p+1} \varphi(A)[H, \ldots, H]) \right. \\ &\quad \left. \times (\nabla^{q+1} \varphi(A)[H, \ldots, H]) \right] \end{aligned} \tag{20}$$

for all $A \in \mathbb{S}$ and symmetric $n \times n$ matrices H. Moreover, we have the following estimate for $m \geq 0$:

$$\left\| \nabla^{m+1} \varphi(A) \right\| \leq (\sqrt{n})^m (m+1)! C_m 2^{-2(m+1)} \lambda_{\min}(A)^{-(m+1/2)}, \tag{21}$$

where $\lambda_{\min}(A) > 0$ is the smallest eigenvalue of A and $C_m := \frac{1}{m+1} \binom{2m}{m}$.

We find that $\partial_l A^{1/2}(x) = \nabla \varphi(A(x))[\partial_l A(x)]$ and

$$\partial_j \partial_l A^{1/2}(x) = \nabla^2 \varphi(A(x))[\partial_j A(x), \partial_l A(x)] + \nabla \varphi(A(x))[\partial_j \partial_l A(x)].$$

Thus, it follows from Eq. (21) that

$$\left\| \partial_l A^{1/2}(x) \right\| \leq C \lambda_{\min}(A(x))^{-1/2} \left\| \partial_l A(x) \right\|,$$

and

$$\begin{aligned} \left\| \partial_j \partial_l A^{1/2}(x) \right\| &\leq C_1 \lambda_{\min}(A(x))^{-(1+1/2)} \left\| \partial_j A(x) \right\| \left\| \partial_l A(x) \right\| \\ &\quad + C_2 \lambda_{\min}(A(x))^{-1/2} \left\| \partial_j \partial_l A(x) \right\|. \end{aligned}$$

More generally, for $m = 1, \ldots, 7$,

$$\begin{aligned} \left\| \partial_{j_1} \ldots \partial_{j_m} A^{1/2}(x) \right\| &\leq C_m \left\{ \sum_{r=0}^{m-1} \lambda_{\min}(A(x))^{-(r+1/2)} \right. \\ &\quad \left. \times \max_{|\beta| \leq m} \left\| \partial_\beta^{|\beta|} A(x) \right\|^{r+1} \right\}. \end{aligned} \tag{22}$$

Let us now provide a lower bound for $\lambda_{\min}(A(x))$ in terms of $tr(A(x))$ and $tr((A(x))^2)$. Define

$$s^2(x) = n^{-1}\left(tr((A(x))^2) - \frac{(tr(A(x)))^2}{n}\right), \quad m(x) = \frac{tr(A(x))}{n}.$$

Then, from Corollary 2.1, Corollary 2.2, and Theorem 2.1 in (111), we obtain

$$\begin{aligned}\frac{1}{\lambda_{\min}(A(x))} &\leq \frac{1}{\lambda_{\max}(A(x))}\left(1 + \frac{2s(x)(n-1)^{1/2}}{m(x) - s(x)(n-1)^{-1/2}}\right) \\ &\leq \frac{1}{m(x) + s(x)(n-1)^{1/2}}\left(1 + \frac{2s(x)(n-1)^{1/2}}{m(x) - s(x)(n-1)^{-1/2}}\right).\end{aligned}$$

Therefore, from Eq. (22), we have for $m = 1, \ldots, 7$ that

$$\left\|\partial_{j_1} \ldots \partial_{j_m} A^{1/2}(x)\right\| \leq C l_m(x), \tag{23}$$

where the function $l_m(x)$ is defined as

$$\begin{aligned}l_m(x) &:= \sum_{r=0}^{m-1}\left(\frac{1}{m(x) + s(x)(n-1)^{1/2}}\left(1 + \frac{2s(x)(n-1)^{1/2}}{m(x) - s(x)(n-1)^{-1/2}}\right)\right)^{-(r+1/2)} \\ &\quad \times \max_{|\beta| \leq m}\left\|\partial_\beta^{|\beta|} A(x)\right\|^{r+1}. \end{aligned} \tag{24}$$

□

The following results are key to guarantee that an SDE is a weak approximation of an optimizer.

Proposition B.6 (Proposition 1 (69)). *Let $0 < \eta < 1$. Consider a stochastic process $X_t, t \geq 0$ satisfying the SDE*

$$dX_t = b(X_t)\, dt + \sqrt{\eta}\sigma(X_t)\, dW_t$$

with $X_0 = x \in \mathbb{R}^d$ and b, σ together with their derivatives belong to G. Define the one-step difference $\Delta = X_\eta - x$, and indicate the i-th component of Δ with Δ_i. Then we have

1. $\mathbb{E}\Delta_i = b_i\eta + \frac{1}{2}\left[\sum_{j=1}^d b_j \partial_{e_j} b_i\right]\eta^2 + \mathcal{O}(\eta^3) \quad \forall i = 1, \ldots, d;$
2. $\mathbb{E}\Delta_i\Delta_j = \left[b_i b_j + \sigma\sigma^T_{(ij)}\right]\eta^2 + \mathcal{O}(\eta^3) \quad \forall i, j = 1, \ldots, d;$
3. $\mathbb{E}\prod_{j=1}^s \Delta_{(i_j)} = \mathcal{O}(\eta^3)$ *for all* $s \geq 3, i_j = 1, \ldots, d.$

All functions above are evaluated at x.

Theorem B.7 (Theorem 2 and Proposition 5, (83)). *Let Assumption B.3 hold and let us define $\bar{\Delta} = x_1 - x$ to be the increment in the discrete-time algorithm, and indicate the i-th component of $\bar{\Delta}$ with $\bar{\Delta}_i$. If in addition there exists $K_1, K_2, K_3, K_4 \in G$ so that*

1. $\left|\mathbb{E}\Delta_i - \mathbb{E}\bar{\Delta}_i\right| \leq K_1(x)\eta^2, \quad \forall i = 1, \ldots, d;$
2. $\left|\mathbb{E}\Delta_i\Delta_j - \mathbb{E}\bar{\Delta}_i\bar{\Delta}_j\right| \leq K_2(x)\eta^2, \quad \forall i, j = 1, \ldots, d;$
3. $\left|\mathbb{E}\prod_{j=1}^s \Delta_{i_j} - \mathbb{E}\prod_{j=1}^s \bar{\Delta}_{i_j}\right| \leq K_3(x)\eta^2, \quad \forall s \geq 3, \quad \forall i_j \in \{1, \ldots, d\};$
4. $\mathbb{E}\prod_{j=1}^3 \left|\bar{\Delta}_{i_j}\right| \leq K_4(x)\eta^2, \quad \forall i_j \in \{1, \ldots, d\}.$

Then, there exists a constant C so that for all $k = 0, 1, \ldots, N$ we have

$$\left|\mathbb{E}g(X_{k\eta}) - \mathbb{E}g(x_k)\right| \leq C\eta.$$

B.3 Limitations

Modeling of discrete-time algorithms using SDEs relies on Assumption B.3. As noted by (71), the approximation can fail when the stepsize η is large or if certain conditions on ∇f and the noise covariance matrix are not met. Although these issues can be addressed by increasing the order of the weak approximation, we believe that the primary purpose of SDEs is to serve as simplification tools that enhance our intuition: We would not benefit significantly from added complexity. Regarding the assumptions on the noise, ours are in line with or more general than those commonly used in the literature.

Another aspect concerns the discretization of SDEs. While our approach has been to experimentally verify that the SDE tracks the evolution of the corresponding discrete algorithms and supports our theoretical insights, alternative theoretical frameworks exist. Notably, backward error analysis offers a promising direction, as it can clarify the role of finite learning rates and help identify different optimizers' implicit biases. This approach has been successfully used to derive higher-order modified equations for SGD (100) and Adam (16). While our work does not include such an analysis, many influential papers (61, 83, 122) similarly omit it. Given that most papers modeling optimizers with SDEs either lack experimental validation or restrict it to artificial landscapes, we take an extra step by validating our insights across various deep neural networks and datasets. To our knowledge, only (88, 20) have conducted experiments involving neural networks, and even then, with relatively small models. In contrast, our extensive experiments demonstrate that our insights apply to realistic scenarios, as confirmed by our numerical results.

Finally, while SDEs benefit from Itô Calculus which allows us to study general non-convex loss functions, we had to focus on simple noise structures. Differently, Stochastic Approximation enables a more fine-grained and insightful analysis for very general noise structures (e.g. multiplicative noise), but often forces the analysis to focus on quadratic losses (90).

B.4 Distributed SGD

This subsection provides the first formal derivation of an SDE model for DSGD. Let us consider the stochastic process $X_t \in \mathbb{R}^d$ defined as the solution of

$$dX_t = -\nabla f(X_t)dt + \sqrt{\frac{\eta}{N}}\sqrt{\hat{\Sigma}(X_t)}dW_t, \tag{25}$$

where $\hat{\Sigma}(x) \coloneqq \frac{1}{N}\sum_{i=1}^{N}\Sigma_i(x)$ is the average of the covariance matrices of the N agents.

Theorem B.8 (Stochastic modified equations)**.** *Let $0 < \eta < 1, T > 0$ and set $N = \lfloor T/\eta \rfloor$. Let $x_k \in \mathbb{R}^d, 0 \leq k \leq N$ denote a sequence of DSGD iterations defined by Eq.* (1)*. Consider the stochastic process X_t defined in Eq.* (25) *and fix some test function $g \in G$ and suppose that g and its partial derivatives up to order 6 belong to G.*
Then, under the assumptions of Section B.1, there exists a constant $C > 0$ independent of η such that for all $k = 0, 1, \ldots, N$, we have

$$|\mathbb{E}g\left(X_{k\eta}\right) - \mathbb{E}g\left(x_k\right)| \leq C\eta.$$

That is, the SDE (25) *is an order* 1 *weak approximation of the DSGD iterations* (1)*.*

Proof. First, we calculate the expected value of the increments of DSGD:

$$\mathbb{E}\left[x_{k+1} - x_k\right] = \mathbb{E}\left[-\frac{\eta}{N}\sum_{i=1}^{N}\nabla f_{\gamma_i}(x_k)\right] = -\eta\nabla f(x_k); \tag{26}$$

Then, we calculate the covariance matrix of the gradient noise of DSGD:

$$\tilde{\Sigma}(x_k) = \eta^2 \mathbb{E}\left[\left(\nabla f(x_k) - \frac{1}{N}\sum_{i=1}^{N} \nabla f_{\gamma_i}(x_k)\right)\left(\nabla f(x_k) - \frac{1}{N}\sum_{j=1}^{N} \nabla f_{\gamma_j}(x_k)\right)^\top\right] \tag{27}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i,j=1}^{N} \mathbb{E}\left[\left(\nabla f(x_k) - \nabla f_{\gamma_i}(x_k)\right)\left(\nabla f(x_k) - \nabla f_{\gamma_j}(x_k)\right)^\top\right] \tag{28}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}\left[\left(\nabla f(x_k) - \nabla f_{\gamma_i}(x_k)\right)\left(\nabla f(x_k) - \nabla f_{\gamma_i}(x_k)\right)^\top\right] \tag{29}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N} \Sigma_i(x_k), \tag{30}$$

where we use independence of $(\nabla f(x_k) - \nabla f_{\gamma_i}(x_k)$ for $i \in [N]$. The thesis follows from Proposition B.6 and Theorem B.7 as drift and diffusion terms are regular by assumption. □

Theorem B.9. *If f is μ-PL, L-smooth, and $Tr(\Sigma_i(x)) < \mathcal{L}_{\sigma_i}$*

$$\mathbb{E}\left[f(X_t) - f(X_*)\right] \leq \left(f(X_0) - f(X_*)\right) e^{-2\mu t} + \frac{\eta L \overline{\mathcal{L}}_\sigma}{4\mu N}(1 - e^{-2\mu t}), \tag{31}$$

where $\overline{\mathcal{L}}_\sigma \coloneqq \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\sigma_i}$.

Proof. Using Ito's Lemma

$$d(f(X_t) - f(X_*)) = -\nabla f(X_t)^\top \nabla f(X_t) dt + \mathcal{O}(\text{Noise}) + \frac{\eta}{2N}\text{Tr}(\nabla^2 f(X_t)\tilde{\Sigma}(X_t))dt \tag{32}$$

$$\leq -2\mu(f(X_t) - f(X_*))dt + \frac{\eta L \overline{\mathcal{L}_\sigma}}{2N} dt + \mathcal{O}(\text{Noise}), \tag{33}$$

which implies the thesis. □

Corollary B.10. *Let the batch size be δB, learning rate $\kappa\eta$, and αN agents. The scaling rule to preserve the performance independently of δ, κ, and α is $\frac{\kappa}{\alpha\delta} = 1$.*

Proof. It follows the same steps as Theorem B.8 to derive the SDE and Theorem B.9 to derive the bound. Then, one needs to find the functional relationship between κ, α, and δ such that the bound does not depend on them. □

Theorem B.11. *If f is L-smooth, we use a learning rate scheduler η_t such that $\phi_t^i = \int_0^t (\eta_s)^i ds$, $\phi_t^1 \overset{t\to\infty}{\to} \infty$, $\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0$, and $\overline{\mathcal{L}}_\sigma \coloneqq \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_{\sigma_i}$*

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right] \leq \frac{f(X_0) - f(X_*)}{\phi_t^1} + \frac{\eta L \overline{\mathcal{L}}_\sigma}{2N}\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0, \tag{34}$$

where $\tilde{t}$ has distribution $\frac{\eta_{\tilde{t}}}{\phi_t^1}$.

Proof. Using Ito's Lemma and using a learning rate scheduler η_t during the derivation of the SDE of Theorem B.8, we have

$$d(f(X_t) - f(X_*)) = -\eta_t \|\nabla f(X_t)\|_2^2 dt + \mathcal{O}(\text{Noise}) + (\eta_t)^2 \frac{\eta}{2N}\text{Tr}(\nabla^2 f(X_t)\tilde{\Sigma}(X_t))dt \tag{35}$$

$$\leq -\eta_t \|\nabla f(X_t)\|_2^2 dt + \mathcal{O}(\text{Noise}) + (\eta_t)^2 \frac{\eta L \overline{\mathcal{L}}_\sigma}{2N} dt. \tag{36}$$

Let us now observe that since $\int_0^t \frac{\eta_s}{\phi_t^1} ds = 1$, the function $s \mapsto \frac{\eta_s}{\phi_t^1}$ defines a probability distribution and let $\tilde{t}$ have that distribution. Then by integrating over time and by the Law of the Unconscious Statistician, we have that

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right] = \frac{1}{\phi_t^1} \int_0^t \|\nabla f(X_s)\|_2^2 \eta_s ds, \tag{37}$$

meaning that

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right] \leq \frac{f(X_0) - f(X_*)}{\phi_t^1} + \frac{\eta L \overline{\mathcal{L}}_\sigma}{2N} \frac{\phi_t^2}{\phi_t^1} \overset{t \to \infty}{\to} 0. \tag{38}$$

□

C Distributed Compressed SGD with Unbiased Compression

This subsection provides the first formal derivation of an SDE model for DCSGD. Let us consider the stochastic process $X_t \in \mathbb{R}^d$ defined as the solution of

$$dX_t = -\nabla f(X_t)dt + \sqrt{\frac{\eta}{N}} \sqrt{\tilde{\Sigma}(X_t)} dW_t, \tag{39}$$

where for $\Phi_{\xi_i,\gamma_i}(x) := \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f_{\gamma_i}(x)$

$$\tilde{\Sigma}(x) = \frac{1}{N} \sum_{i=1}^N \left(\mathbb{E}_{\xi_i \gamma_i} \left[\Phi_{\xi_i,\gamma_i}(x) \Phi_{\xi_i,\gamma_i}(x)^\top \right] + \Sigma_i(x) \right). \tag{40}$$

Before proceeding, we ensure that the SDE admits a unique solution and that its coefficients are sufficiently regular.

Lemma C.1. *The drift term ∇f is Lipschitz, satisfies Affine Growth, and is in G together with all its derivatives.*

Proof. This is obvious as we assume all of these conditions. □

Regarding the diffusion term, we have that

Lemma C.2. *The diffusion term $\tilde{\Sigma}(x)$ satisfies Affine Growth.*

Proof. Since $\|\sqrt{\tilde{\Sigma}_i}(x)\|_2 \leq \mathrm{Tr}(\tilde{\Sigma}_i(x))^{\frac{1}{2}} \leq (\omega \|\nabla f(x)\|_2^2 + \|\Sigma_i(x)\|_\infty(\omega+1))^{\frac{1}{2}}$, the linear growth of the gradient, the boundedness of Σ_i, and that $\|A\|_\infty \leq \sqrt{d}\|A\|_2$ for each matrix A. □

Lemma C.3. *Let us assume the same assumptions as Lemma B.4 and that the domain is closed and sufficiently large.*[6] *Additionally, assume that*

$$\sup_{x \in K} |g(x,\cdot)| \in L^1(\mathbb{R}^n)$$

for all compact sets K. Then the entries of $\tilde{\Sigma}$ in Eq. 39 are in $C_b^7(\mathbb{R}^n)$.

Proof. Since we are on a closed and sufficiently large domain, by the definition of $\tilde{\Sigma}$, dominated convergence, and from the additional assumption on g, it follows that $\tilde{\Sigma}$ is continuous. So Lemma B.4 entails that the entries of $\tilde{\Sigma}$ are in $C_b^7(\mathbb{R}^n)$. □

Lemma C.4. *The diffusion term $\tilde{\Sigma}(x)$ is Definite Positive.*

Proof. By the definition of $\tilde{\Sigma}(x)$ and the fact that $\Sigma_i(x)$ are DP by assumption, the thesis follows. □

Corollary C.5. *Since $\tilde{\Sigma}$ is positive definite and its entries are in $C_b^7(\mathbb{R}^n)$, $\sqrt{\tilde{\Sigma}}$ is Lipschitz.*

Proof. The function

$$\varphi : \mathbb{S} \to \mathbb{S}, \quad A \mapsto \sqrt{A}$$

has Fréchet derivatives of any order on $\mathbb{S}$ (see e.g. (28)). Therefore, $\tilde{\Sigma}^{1/2} \in C^7(\mathbb{R}^n)$, and since $\tilde{\Sigma} \in C_b^7(\mathbb{R}^n)$, $\tilde{\Sigma}^{1/2}$ is Lipschitz continuous (see Proposition 6.2 in (50)). □

[6] This is a common assumption in the literature (98, 108, 121, 117, 6, 7, 80, 30).

Lemma C.6. *Under the same assumptions as Lemma B.5,* $\tilde{\Sigma}^{1/2} \in G$ *together with its derivatives.*

Proof. The thesis follows from the regularity of the entries and the closeness and boundedness of the domain. □

Remark C.7. Based on the above results, we have that under mild assumptions on the noise structures (see Sec. B.1) that cover and generalize the well-accepted Gaussianity, and under the well-accepted closeness and boundedness of the domain, the SDE of DCSGD admits a unique solution and its coefficients are regular enough to apply Prop. B.6 and Thm. B.7.

Theorem C.8 (Stochastic modified equations)**.** *Let* $0 < \eta < 1, T > 0$ *and set* $N = \lfloor T/\eta \rfloor$*. Let* $x_k \in \mathbb{R}^d, 0 \leq k \leq N$ *denote a sequence of DCSGD iterations defined by Eq.* (5)*. Consider the stochastic process* X_t *defined in Eq.* (39) *and fix some test function* $g \in G$ *and suppose that* g *and its partial derivatives up to order 6 belong to* G*.*
Then, under the assumptions of Section B.1, there exists a constant $C > 0$ *independent of* η *such that for all* $k = 0, 1, \ldots, N$*, we have*

$$|\mathbb{E}g\left(X_{k\eta}\right) - \mathbb{E}g\left(x_k\right)| \leq C\eta.$$

That is, the SDE (39) *is an order* 1 *weak approximation of the DCSGD iterations* (5)*.*

Proof. First, we calculate the expected value of the increments of DCSGD:

$$\mathbb{E}\left[x_{k+1} - x_k\right] = \mathbb{E}\left[-\frac{\eta}{N}\sum_{i=1}^{N}\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right)\right] = \mathbb{E}\left[-\frac{\eta}{N}\sum_{i=1}^{N}\nabla f_{\gamma_i}(x_k)\right] \tag{41}$$

$$= -\frac{\eta}{N}\sum_{i=1}^{N}\nabla f(x_k) = -\eta\nabla f(x_k); \tag{42}$$

Then, we calculate the covariance matrix of the gradient noise of DCSGD:

$$\tilde{\Sigma}(x_k) = \eta^2\mathbb{E}_{\xi\gamma}\left[\left(\nabla f(x_k) - \frac{1}{N}\sum_{i=1}^{N}\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right)\right)\left(\nabla f(x_k) - \frac{1}{N}\sum_{j=1}^{N}\mathcal{C}_{\xi_j}\left(\nabla f_{\gamma_j}(x_k)\right)\right)^\top\right] \tag{43}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i,j=1}^{N}\mathbb{E}_{\xi_i\xi_j\gamma_i\gamma_j}\left[\left(\nabla f(x_k) - \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right)\right)\left(\nabla f(x_k) - \mathcal{C}_{\xi_j}\left(\nabla f_{\gamma_j}(x_k)\right)\right)^\top\right] \tag{44}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\xi_i\gamma_i}\left[\left(\nabla f(x_k) - \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right)\right)\left(\nabla f(x_k) - \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x_k)\right)\right)^\top\right] \tag{45}$$

$$= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N}\left(\mathbb{E}_{\xi_i\gamma_i}\left[\Phi_{\xi_i,\gamma_i}(x_k)\Phi_{\xi_i,\gamma_i}(x_k)^\top\right] + \Sigma_i(x_k)\right), \tag{46}$$

where $\Phi_{\xi_i,\gamma_i}(x) := \mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f_{\gamma_i}(x)$ and we use independence of $\mathcal{C}_{\xi_i}$ and $\nabla f(x_k) - \nabla f_{\gamma_i}(x_k)$ for all $i \in [N]$. Remembering Remark C.7, the thesis follows from Prop. B.6 and Thm. B.7. □

Remark C.9. The expression for $\tilde{\Sigma}(x)$ is easily derived for different compressors by leveraging Proposition 21 in (90).

In all the following results, the reader will notice that all the drifts, diffusion terms, and noise assumptions are selected to guarantee that the SDE we derived for DCSGD is indeed a 1 weak approximation for DCSGD.

Theorem C.10. *If* f *is* μ*-PL,* L*-smooth,* $\overline{\omega} = \frac{1}{N}\sum_{i=1}^{N}\omega_i$*,* $Tr(\Sigma_i(x)) < \mathcal{L}_{\sigma_i}$*,* $\overline{\mathcal{L}}_\sigma := \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\sigma_i}$*, and* $\overline{\omega\mathcal{L}_\sigma} := \frac{1}{N}\sum_{i=1}^{N}\omega_i\mathcal{L}_{\sigma_i}$

$$\mathbb{E}\left[f(X_t) - f(X_*)\right] \leq (f(X_0) - f(X_*))e^{-\left(2\mu - \frac{\eta L^2\overline{\omega}}{N}\right)t} + \left(1 - e^{-\left(2\mu - \frac{\eta L^2\overline{\omega}}{N}\right)t}\right)\frac{\frac{\eta L\left(\overline{\mathcal{L}}_\sigma + \overline{\omega\mathcal{L}_\sigma}\right)}{2N}}{\left(2\mu - \frac{\eta L^2\overline{\omega}}{N}\right)}. \tag{47}$$

Proof. Using Ito's Lemma

$$d(f(X_t) - f(X_*)) = -\nabla f(X_t)^\top\nabla f(X_t)dt + \mathcal{O}(\text{Noise}) + \frac{\eta}{2N}\text{Tr}(\nabla^2 f(X_t)\tilde{\Sigma}(X_t))dt \tag{48}$$

$$\leq -2\mu(f(X_t) - f(X_*))dt \tag{49}$$

$$+ \frac{\eta L}{2N}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{\xi_i,\gamma_i}\|(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f(x))\|_2^2\right)dt + \mathcal{O}(\text{Noise}). \tag{50}$$

Let us focus on a single element of the summation:

$$\mathbb{E}_{\xi_i,\gamma_i}\|(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right)-\nabla f(x))\|_2^2 \quad = \mathbb{E}_{\gamma_i}\left[\mathbb{E}_{\xi_i}\left[\|\mathcal{C}_{\xi_i}(\nabla f_{\gamma_i}(x))-\nabla f_{\gamma_i}(x)\|^2+\|\nabla f_{\gamma_i}(x)-\nabla f(x)\|^2\right]\mid\gamma_i\right] \tag{51}$$

$$\leq \omega_i\mathbb{E}_{\gamma_i}\|\nabla f_{\gamma_i}(x)\|_2^2+\mathbb{E}_{\gamma_i}\left[\|\nabla f_{\gamma_i}(x)-\nabla f(x)\|^2\right]=\omega_i\|\nabla f(x)\|_2^2+(\omega_i+1)\mathbb{E}_{\gamma_i}\left[\|\nabla f_{\gamma_i}(x)-\nabla f(x)\|^2\right] \tag{52}$$

$$\leq 2\omega_i L(f(x)-f(x_*))+\mathcal{L}_{\sigma_i}(\omega_i+1). \tag{53}$$

Therefore, we have that

$$d(f(X_t)-f(X_*))\leq -2\mu(f(X_t)-f(X_*))dt+\mathcal{O}(\text{Noise})+\frac{\eta L^2\overline{\omega}}{N}(f(X_t)-f(X_*))dt \tag{54}$$

$$+\frac{\eta L\left(\overline{\mathcal{L}_\sigma}+\overline{\omega\mathcal{L}_\sigma}\right)}{2N}dt, \tag{55}$$

which implies the thesis. □

Remark: We observe that $\overline{\omega\mathcal{L}_\sigma}$ gives a tighter bound than $\overline{\omega}\mathcal{L}_{\sigma,\max}$ or $\omega_{\max}\overline{\mathcal{L}}_\sigma$.

Theorem C.11. *If f is L-smooth, we use a learning rate scheduler η_t such that $\phi_t^i=\int_0^t(\eta_s)^i ds$, $\phi_t^1\overset{t\to\infty}{\to}\infty$, $\frac{\phi_t^2}{\phi_t^1}\overset{t\to\infty}{\to}0$, and $\eta_t>\frac{\eta L\overline{\omega}}{2N}(\eta_t)^2$, then,*

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right]\leq\frac{1}{1-\frac{\eta L\overline{\omega}}{2N}\frac{\phi_t^2}{\phi_t^1}}\left(\frac{f(X_*)-f(X_0)}{\phi_t^1}+\frac{\phi_t^2}{\phi_t^1}\frac{\eta L}{2N}\left(\overline{\mathcal{L}_\sigma}+\overline{\omega\mathcal{L}_\sigma}\right)\right)\overset{t\to\infty}{\to}0, \tag{56}$$

where $\tilde{t}$, is a random time with distribution $\frac{\eta_{\tilde{t}}-\frac{\eta L\overline{\omega}}{2N}(\eta_{\tilde{t}})^2}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}$.

Proof. Leveraging what we have shown above, we have that

$$d(f(X_t)-f(X_*))=-\eta_t\|\nabla f(X_t)\|_2^2dt+\mathcal{O}(\text{Noise}) \tag{57}$$

$$+(\eta_t)^2\frac{\eta L}{2N}\left(\frac{1}{N}\sum_{i=1}^N\mathbb{E}_{\xi_i,\gamma_i}\|(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right)-\nabla f(x))\|_2^2\right)dt. \tag{58}$$

As before, $\mathbb{E}_{\xi_i,\gamma_i}\|(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right)-\nabla f(x))\|_2^2\leq\omega_i\|\nabla f(x)\|_2^2+\mathcal{L}_{\sigma_i}(\omega_i+1)$. Therefore, we have that

$$\mathbb{E}\left[\|\nabla f(X_t)\|_2^2\right]\left(\eta_t-\frac{\eta L\overline{\omega}}{2N}(\eta_t)^2\right)dt\leq -d(f(X_t)-f(X_*))+\frac{\eta L(\eta_t)^2}{2N}\left(\overline{\mathcal{L}_\sigma}+\overline{\omega\mathcal{L}_\sigma}\right)dt. \tag{59}$$

Let us now observe that since $\int_0^t\frac{\eta_s-\frac{\eta L\overline{\omega}}{2N}\eta_s^2}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}ds=1$, the function $s\mapsto\frac{\eta_s-\frac{\eta L\overline{\omega}}{2N}\eta_s^2}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}$ defines a probability distribution and let $\tilde{t}$ have that distribution. Then by integrating over time and by the Law of the Unconscious Statistician, we have that

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right]=\frac{1}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}\int_0^t\|\nabla f(X_s)\|_2^2\left(\eta_s-\frac{\eta L\overline{\omega}}{2N}\eta_s^2\right)ds, \tag{60}$$

meaning that

$$\mathbb{E}\left[\|\nabla f(X_{\tilde{t}})\|_2^2\right]\leq\frac{1}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}\left(f(X_*)-f(X_0)+\phi_t^2\frac{\eta L}{2N}\left(\overline{\mathcal{L}_\sigma}+\overline{\omega\mathcal{L}_\sigma}\right)\right)\overset{t\to\infty}{\to}0, \tag{61}$$

where $\tilde{t}$, is a random time with distribution $\frac{\eta_{\tilde{t}}-\frac{\eta L\overline{\omega}}{2N}(\eta_{\tilde{t}})^2}{\phi_t^1-\frac{\eta L\overline{\omega}}{2N}\phi_t^2}$. □

C.1 Scaling Rules: Recovering DSGD

Proposition C.12. *Let the batch size be δB, learning rate $\kappa\eta$, the compression rates $\beta\omega_i$, and αN agents. The scaling rules to recover the performance of DSGD are complex and many. For practicality and interpretability purposes, we list here those involving only two hyperparameters at the time:*

1. *If $\kappa = \delta = 1$, one needs to ensure that the relation between α and β is*

$$\alpha = 1 + \beta \left(\frac{\overline{\omega \mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}} + \frac{\overline{\omega} \overline{\mathcal{L}_\sigma} \eta L^2}{2\mu N} \right). \tag{62}$$

This gives rise to a trade-off between agents and compression: If there is compression, then one needs to increase the number of agents, and the stronger the compression, the more is needed. In the absence of compression, no additional agents are needed.

2. *If $\beta = \delta = 1$, one needs to ensure that the relation between α and κ is*

$$\frac{\alpha}{\kappa} = 1 + \frac{\overline{\omega \mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}} + \frac{\overline{\omega} \overline{\mathcal{L}_\sigma} \eta L^2}{2\mu N}. \tag{63}$$

This gives rise to a trade-off between agents and learning rate: If there is compression, one can increase the learning rate, i.e. $\kappa > 1$, and compensate with more agents $\alpha > \kappa > 1$. If no compression is in place, the classic trade-off of DSGD $\alpha = \kappa$ is recovered.

3. *If $\beta = \kappa = 1$, one needs to ensure that the relation between α and γ is*

$$\alpha = \frac{1 + \frac{\overline{\omega \mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}}}{\delta} + \frac{\overline{\omega} \eta L^2}{2\mu N}. \tag{64}$$

This gives rise to a trade-off between agents and batch size: If there is compression, one can increase the batch size, i.e. $\delta \geq 1$, and needs fewer agents. If no compression is in place, the classic trade-off of DSGD $\alpha\delta = 1$ is recovered.

4. *If $\alpha = \delta = 1$, one needs to ensure that the relation between β and κ is*

$$\kappa = \frac{\overline{\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma} + \beta \left(\overline{\omega \mathcal{L}_\sigma} + \frac{\overline{\omega} \eta L^2}{2\mu N} \right)}. \tag{65}$$

This gives rise to a trade-off between learning rate and compression: More compression, requires a lower learning rate. No compression implies no change in the learning rate.

5. *If $\alpha = \kappa = 1$, one needs to ensure that the relation between β and δ is*

$$\delta = \frac{2\mu \left(\overline{\mathcal{L}_\sigma} + \beta \overline{\omega \mathcal{L}_\sigma} \right)}{\overline{\mathcal{L}_\sigma} \left(2\mu - \beta \frac{\overline{\omega} \eta L^2}{N} \right)}. \tag{66}$$

This gives rise to a trade-off between batch size and compression: More compression, requires a larger batch size. No compression implies no change in batch size.

6. *If $\alpha = \beta = 1$, one needs to ensure that the relation between κ and δ is*

$$\kappa = \frac{\delta \overline{\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma} + \overline{\omega \mathcal{L}_\sigma} + \delta \frac{\overline{\omega} \eta L^2}{2\mu N}}. \tag{67}$$

This gives rise to a trade-off between learning rate and batch size: More batch size requires a larger learning rate. No compression implies the classic $\kappa = \delta$ of DSGD.

We summarize the derived rules in the following table: Of course, in the absence of compression, all scaling rules reduce to the scaling rules of DSGD.

Proof. Using Ito on f, we have that

$$d(f(X_t) - f(X_*)) = -\kappa \|\nabla f(X_t)\|_2^2 dt + \mathcal{O}(\text{Noise}) + \frac{\eta \kappa^2}{2\alpha N} \text{Tr}(\nabla^2 f(X_t) \tilde{\Sigma}(X_t)) dt \tag{68}$$

$$\leq -2\mu\kappa (f(X_t) - f(X_*)) dt + \mathcal{O}(\text{Noise}) \tag{69}$$

$$+ \frac{\eta \kappa^2 L}{2\alpha N} \frac{1}{\alpha N} \sum_{i=1}^{\alpha N} \mathbb{E}_{\xi_i, \gamma_i} \| (\mathcal{C}_{\xi_i} (\nabla f_{\gamma_i}(x)) - \nabla f(x)) \|_2^2 dt. \tag{70}$$

Scaling Rule	Implication
$\alpha = 1 + \beta\frac{\overline{\omega\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}} + \beta\frac{\overline{\omega}\overline{\mathcal{L}_\sigma}\eta L^2}{2\mu N}$	CR $\uparrow \implies$ Agents $\uparrow$
$\frac{\alpha}{\kappa} = 1 + \frac{\overline{\omega\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}} + \frac{\overline{\omega}\overline{\mathcal{L}_\sigma}\eta L^2}{2\mu N}$	LR $\uparrow \implies$ Agents $\uparrow$
$\alpha = \frac{1}{\delta}\left(1 + \frac{\overline{\omega\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma}}\right) + \frac{\overline{\omega}\eta L^2}{2\mu N}$	BS $\downarrow \implies$ Agents $\uparrow$
$\kappa = \frac{\overline{\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma} + \beta\left(\overline{\omega\mathcal{L}_\sigma} + \frac{\overline{\omega}\eta L^2}{2\mu N}\right)}$	CR $\uparrow \implies$ LR $\downarrow$
$\delta = \frac{2\mu\left(\overline{\mathcal{L}_\sigma} + \beta\overline{\omega\mathcal{L}_\sigma}\right)}{\overline{\mathcal{L}_\sigma}\left(2\mu - \beta\frac{\overline{\omega}\eta L^2}{N}\right)}$	CR $\uparrow \implies$ BS $\uparrow$
$\kappa = \frac{\delta\overline{\mathcal{L}_\sigma}}{\overline{\mathcal{L}_\sigma} + \overline{\omega\mathcal{L}_\sigma} + \delta\frac{\overline{\omega}\eta L^2}{2\mu N}}$	BS $\uparrow \implies$ LR $\uparrow$

Table 2: Summary of Trade-offs Between Parameters (CR = Compression Rate, LR = Learning Rate, and BS = Batch Size).

As above, $\mathbb{E}_{\xi_i,\gamma_i}\|(\mathcal{C}_{\xi_i}(\nabla f_{\gamma_i}(x)) - \nabla f(x))\|_2^2 \leq 2\beta\omega_i L(f(x) - f(x_*)) + \frac{\mathcal{L}_{\sigma_i}}{\delta B}(\beta\omega_i + 1)$. Therefore, we have that

$$d(f(X_t) - f(X_*)) \tag{71}$$

$$\leq -2\mu\kappa(f(X_t) - f(X_*))dt + \mathcal{O}(\text{Noise}) + \frac{\eta\kappa^2 L^2\overline{\omega}\beta}{\alpha N}(f(X_t) - f(X_*))dt \tag{72}$$

$$+\frac{\eta\kappa^2 L\overline{\mathcal{L}}_\sigma}{2B\delta\alpha N}dt + \frac{\beta\eta\kappa^2 L\overline{\omega\mathcal{L}_\sigma}}{2B\delta\alpha N}dt, \tag{73}$$

which implies that

$$\mathbb{E}\left[f(X_t) - f(X_*)\right] \leq (f(X_0) - f(X_*))e^{-\left(2\mu - \frac{\eta\kappa L^2\overline{\omega}\beta}{\alpha N}\right)\kappa t} \tag{74}$$

$$+\left(1 - e^{-\kappa\left(2\mu - \frac{\eta\kappa L^2\overline{\omega}}{\alpha N}\right)t}\right)\frac{\frac{\eta\kappa L\overline{\mathcal{L}}_\sigma}{2B\delta\alpha N} + \frac{\beta\eta\kappa L\overline{\omega\mathcal{L}_\sigma}}{2B\delta\alpha N}}{\left(2\mu - \frac{\eta\kappa L^2\overline{\omega}\beta}{\alpha N}\right)}. \tag{75}$$

Now, we need to find functional relationships between α, δ, κ, and β such that the asymptotic value of the loss of DCSGD with hyperparameters $(\kappa\eta, \delta B, \beta\omega_i, \alpha N)$ matches the asymptotic loss value of DSGD with hyperparameters (η, B, N):

$$\frac{\frac{\eta\kappa L\overline{\mathcal{L}}_\sigma}{2B\delta\alpha N} + \frac{\beta\eta\kappa L\overline{\omega\mathcal{L}_\sigma}}{2B\delta\alpha N}}{\left(2\mu - \frac{\eta\kappa L^2\overline{\omega}\beta}{\alpha N}\right)} = \frac{\eta L\overline{\mathcal{L}}_\sigma}{4\mu NB}. \tag{76}$$

Since a general formula involving all four quantities is difficult to interpret, we derive six rules: For each of them, we keep two scaling parameters constant to 1 and study the relationship between the remaining two.

Let us prove one to show the mechanism as they are all derived in a few passages. We focus on the first one, for which we set $\kappa = \delta = 1$ and study the relationship between α and β. To do this, we solve

$$\frac{\frac{\eta L\overline{\mathcal{L}}_\sigma}{2B\alpha N} + \frac{\beta\eta L\overline{\omega\mathcal{L}_\sigma}}{2B\alpha N}}{\left(2\mu - \frac{\eta L^2\overline{\omega}\beta}{\alpha N}\right)} = \frac{\eta L\overline{\mathcal{L}}_\sigma}{4\mu NB} \implies \frac{\frac{\overline{\mathcal{L}}_\sigma + \beta\overline{\omega\mathcal{L}_\sigma}}{\alpha}}{\left(2\mu - \frac{\eta L^2\overline{\omega}\beta}{\alpha N}\right)} = \frac{\overline{\mathcal{L}}_\sigma}{2\mu} \tag{77}$$

$$\implies \frac{1}{\alpha}2\mu\left(\overline{\mathcal{L}}_\sigma + \beta\overline{\omega\mathcal{L}_\sigma}\right) = \overline{\mathcal{L}}_\sigma\left(2\mu - \frac{1}{\alpha}\frac{\eta L^2\overline{\omega}\beta}{N}\right), \tag{78}$$

which implies the thesis. All the other rules are derived similarly. □

C.2 Stationary Distribution

Let us focus on a quadratic function $f(x) = \frac{x^\top Hx}{2}$ such that $H = \text{diag}(\lambda_1, \cdots, \lambda_d)$ where each $\lambda_j > 0$.

Proposition C.13. *Let us consider the k-Sparsification compressor. Then,*

$$\mathbb{E}[X_t] = e^{-Ht}X_0 \to 0, \tag{79}$$

and, for $M := 2H\left(I_d - \frac{\eta H}{2N}\left(\frac{d}{k} - 1\right)\right)$,

$$Cov\left[X_t\right] = e^{-Mt}X_0^2 + \frac{\eta}{N}\frac{d}{k}\overline{\sigma^2}M^{-1}(I_d - e^{-Mt}) - e^{-2Ht}X_0^2 \to \frac{\eta}{N}\frac{d}{k}\overline{\sigma^2}M^{-1}. \tag{80}$$

Proof. It is clear that

$$d\mathbb{E}[X_t] = -H\mathbb{E}[X_t]dt, \tag{81}$$

which implies that

$$\mathbb{E}[X_t] = e^{-Ht}X_0 \to 0. \tag{82}$$

Let us now focus on the dynamics of the square of the j-th coordinate $(X_t)_j$ of X_t which, for ease of notation, we call Z_t. Since we need to apply Ito's Lemma on $((X_t)_j)^2$, we need to observe that since this is the square of j-th component of X_t, it can be rewritten as the square of the projection of X_t on the j-th coordinate as $\pi_j(X_t)$. Therefore, we have that by Ito's Lemma:

$$d((\pi_j(X_t))^2) = \partial_t((\pi_j(X_t))^2)dt + \langle\nabla((\pi_j(X_t))^2), \nabla f(X_t)\rangle dt \tag{83}$$

$$+ \frac{1}{2}\mathrm{Tr}\left(\frac{\eta}{N}\tilde{\Sigma}(X_t)\nabla^2((\pi_j(X_t))^2)\right)dt + \langle\nabla(\pi_j(X_t))^2, \sigma_t\rangle dW_t \tag{84}$$

$$= -(HX_t)^\top\nabla((\pi_j(X_t))^2)dt + \mathcal{O}(\text{Noise}) + \frac{\eta}{2N}\mathrm{Tr}\left(\nabla^2((\pi_j(X_t))^2)\tilde{\Sigma}(X_t)\right)dt. \tag{85}$$

Since $\nabla((\pi_j(X_t))^2) = (0, \cdots, 0, 2(X_t)_j, 0, \cdots, 0)$ and $\nabla^2((\pi_j(X_t))^2) = \mathrm{diag}\,(0, \cdots, 0, 2, 0, \cdots, 0)$, we have that

$$d\left(((X_t)_j)^2\right) = d\left((\pi_j(X_t))^2\right) = -(HX_t)^\top\nabla((\pi_j(X_t))^2)dt + \mathcal{O}(\text{Noise}) \tag{86}$$

$$+ \frac{\eta}{2N}\mathrm{Tr}\left(\nabla^2((\pi_j(X_t))^2)\tilde{\Sigma}(X_t)\right)dt, \tag{87}$$

meaning that

$$d(Z_t^2) = -2h_jZ_t^2dt + \mathcal{O}(\text{Noise}) + \frac{\eta}{N}\tilde{\Sigma}_{jj}(X_t)dt. \tag{88}$$

Since we have that

$$\tilde{\Sigma}(x) = \frac{1}{N}\sum_{i=1}^N\mathbb{E}_{\xi_i\gamma_i}\left[\left(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f(x)\right)\left(\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right) - \nabla f(x)\right)^\top\right] \tag{89}$$

$$= \frac{1}{N}\sum_{i=1}^N\mathbb{E}_{\xi_i\gamma_i}\left[\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right)\mathcal{C}_{\xi_i}\left(\nabla f_{\gamma_i}(x)\right)^\top\right] - \nabla f(x)\nabla f(x)^\top \tag{90}$$

$$= \frac{1}{N}\sum_{i=1}^N\left(\frac{d}{k}\nabla f(x)\nabla f(x)^\top + \frac{d}{k}(\Sigma_i(X_t))\right) - \nabla f(x)\nabla f(x)^\top \tag{91}$$

$$= \left(\frac{d}{k} - 1\right)\nabla f(x)\nabla f(x)^\top + \frac{d}{k}\overline{\Sigma^2}, \tag{92}$$

where $\overline{\Sigma^2} := \frac{1}{N}\sum_{i=1}^N(\Sigma_i(X_t))$. Therefore, we have that

$$\mathbb{E}\left[(X_t)^2\right] = e^{-Mt}X_0^2 + \frac{\eta}{N}\frac{d}{k}\overline{\sigma^2}M^{-1}(I_d - e^{-Mt}) \tag{93}$$

where $\overline{\sigma^2} := \mathrm{diag}(\overline{\Sigma^2})$. The thesis follows from here. □

D Distributed SignSGD

This subsection provides the first formal derivation of an SDE model for DSignSGD. Note that the single node case was simultaneously tackled by (21) and (113): The first derived the SDE for SignSGD under the WA framework, while (113) derived an SDE for SignSGD in the high dimensional setting for a linear regression task — See Appendix F in (113) for a comparison between the two derivations. Let us consider the stochastic process $X_t \in \mathbb{R}^d$ defined as the solution of

$$dX_t = -\frac{1}{N}\sum_{i=1}^N\left(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(X_t) < 0)\right)dt + \sqrt{\frac{\eta}{N}}\sqrt{\overline{\Sigma}(X_t)}dW_t. \tag{94}$$

where

$$\overline{\Sigma}(X_t) := \frac{1}{N}\sum_{i=1}^N\overline{\Sigma_i}(X_t), \tag{95}$$

and $\overline{\Sigma_i}(x) = \mathbb{E}[\xi_{\gamma_i}(x)\xi_{\gamma_i}(x)^\top]$ where $\xi_{\gamma_i}(x) := \mathrm{sign}(\nabla f_{\gamma_i}(x)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_i}(x) < 0)$ the noise in the sample $\mathrm{sign}\left(\nabla f_{\gamma_i}(x)\right)$. Before proceeding, we ensure that the SDE admits a unique solution and that its coefficients are sufficiently regular.

Lemma D.1. *The drift term* $b(x) := \frac{1}{N}\sum_{i=1}^{N}(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(x) < 0))$ *is Lipschitz, satisfies affine growth, and belongs to the space* G *together with its derivatives.*

Proof. Since we are assuming that the gradient noise has a smooth and bounded probability density function,[7] the drift can be rewritten in terms of the CDF $F_{Z_i}(x)$ of the noise as the average of $b_i(x) := 1 - 2F_{Z_i}(-\nabla f(x))$, whose derivative is $2F'_{Z_i}(-\nabla f(x))\nabla^2 f(x)$. Since the density functions and the Hessian of f are bounded, we conclude that the derivative is bounded. Therefore, the drift is Lipschitz and as regular as ∇f, meaning that each entry is in G, together with its derivatives. Finally, since it is bounded, it has affine growth. □

Lemma D.2. *The diffusion coefficient* $\sqrt{\overline{\Sigma}}$ *satisfies the affine growth condition.*

Proof. Since it is bounded, the result follows immediately. □

Lemma D.3. *Let us assume the same assumptions as Lemma B.4. Additionally, assume that*

$$\sup_{x\in K}|g(x,\cdot)| \in L^1(\mathbb{R}^n)$$

for all compact sets K*. Then the entries of* $\overline{\Sigma}$ *in Eq. 95 are in* $C_b^7(\mathbb{R}^n)$*.*

Proof. By the definition of $\overline{\Sigma}$ in terms of the sign-function and dominated convergence, from the additional assumption on g, it follows that $\overline{\Sigma}$ is continuous. So Lemma B.4 entails that the entries of $\overline{\Sigma}$ are in $C_b^7(\mathbb{R}^n)$. □

Lemma D.4. *Under the assumption that*

$$g(x,y) > 0, \tag{96}$$

the covariance matrix $\overline{\Sigma}$ *is positive definite.*

Proof. Let us focus on the case $N = 1$ as the generalization is straightforward. For $y = (y_1, \dots, y_n)^T$, observe that

$$\left(\overline{\Sigma}(x)y, y\right) = \sum_{i,j=1}^{n} y_i \mathbb{E}\left[\xi_\gamma^i(x)\xi_\gamma^j(x)\right] y_j = \mathbb{E}\left[\left(\sum_{i=1}^{n}\xi_\gamma^i(x)y_i\right)^2\right].$$

Using the definition of ξ_γ and the positivity of the density g, we can argue by contradiction and see that for $y \neq 0$, the right-hand side of the equation must be strictly greater than zero for all x. Therefore, $\overline{\Sigma}(x) \in \mathbb{S}$ for all x, where $\mathbb{S}$ denotes the open set of positive definite matrices in the space of symmetric $n \times n$ matrices. □

Corollary D.5. *Since* $\overline{\Sigma}$ *is positive definite and its entries are in* $C_b^7(\mathbb{R}^n)$*,* $\sqrt{\overline{\Sigma}}$ *is Lipschitz.*

Proof. The function

$$\varphi : \mathbb{S} \to \mathbb{S}, \quad A \mapsto \sqrt{A}$$

has Fréchet derivatives of any order on $\mathbb{S}$ (see e.g. (28)). Therefore, $\overline{\Sigma}^{1/2} \in C^7(\mathbb{R}^n)$, and since $\overline{\Sigma} \in C_b^7(\mathbb{R}^n)$, $\overline{\Sigma}^{1/2}$ is Lipschitz continuous (see Proposition 6.2 in (50)). □

Proposition D.6. *Assume the conditions of Lemma B.5 and assume that the functions* $l_m(x)$ *for* $m = 1, \dots, 7$ *in Eq. (19) are of polynomial growth. Then* $\overline{\Sigma}^{1/2} \in G$ *together with its derivatives.*

Corollary D.7. *If the noise* $Z(x) \sim \mathcal{N}(0, \Sigma)$ *or* $Z(x) \sim t_\nu(0, \Sigma)$*, then* $\overline{\Sigma}^{1/2} \in G$ *together with its derivatives.*

Proof. With the definition of $\Xi_\nu(x)$ given in Corollary D.10, the function $K(x) := \sqrt{1 - 4\Xi_\nu(x)^2}$ is in G together with its derivative: It is easy to verify that all the derivatives of $K(x)$ are bounded even in the case $\nu = 1$, which is the most pathological one. Therefore, in the case $N = 1$, $\sqrt{\overline{\Sigma}}(x)$ is in G together with its derivatives. Generalizing to $N > 1$ follows the same steps. □

Remark D.8. Based on the above results, we have that under mild assumptions on the noise structures (see Sec. B.1) that cover and generalize the well-accepted Gaussianity, e.g. covering Student's t as well, the SDE of DSignSGD admits a unique solution and its coefficients are regular enough to apply Prop. B.6 and Thm. B.7.

[7] This is commonly assumed in the literature. Among others, (2, 18, 77, 102, 124, 112, 114) assume that it is Gaussian, while (55) offers an intuitive justification.

Theorem D.9 (Stochastic modified equations)**.** *Let $0 < \eta < 1, T > 0$ and set $N = \lfloor T/\eta \rfloor$. Let $x_k \in \mathbb{R}^d, 0 \leq k \leq N$ denote a sequence of DSignSGD iterations defined by Eq.* (10)*. Consider the stochastic process X_t defined in Eq.* (94) *and fix some test function $g \in G$ and suppose that g and its partial derivatives up to order 6 belong to G. Then, under the assumptions of Section B.1, there exists a constant $C > 0$ independent of η such that for all $k = 0, 1, \ldots, N$, we have*

$$|\mathbb{E}g\left(X_{k\eta}\right) - \mathbb{E}g\left(x_k\right)| \leq C\eta.$$

That is, the SDE (94) *is an order* 1 *weak approximation of the DSignSGD iterations* (10)*.*

Proof. First, we calculate the expected value of the increments of DSignSGD:

$$\mathbb{E}\left[x_{k+1} - x_k\right] = \mathbb{E}\left[-\frac{\eta}{N}\sum_{i=1}^{N}\operatorname{sign}(\nabla f_{\gamma_i}(x_k))\right] = -\frac{\eta}{N}\sum_{i=1}^{N}\left(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right); \tag{97}$$

Then, we calculate the covariance matrix of the gradient noise of DSignSGD:

$$\begin{aligned}
\overline{\Sigma}(x_k) = \eta^2\mathbb{E}_\gamma\Bigg[&\left(\frac{1}{N}\sum_{i=1}^{N}\operatorname{sign}(\nabla f_{\gamma_i}(x_k) - \frac{1}{N}\sum_{i=1}^{N}\left(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right)\right) && (98)\\
&\left(\frac{1}{N}\sum_{i=1}^{N}\operatorname{sign}(\nabla f_{\gamma_i}(x_k) - \frac{1}{N}\sum_{i=1}^{N}\left(1 - 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right)\right)^\top\Bigg] && (99)\\
= \frac{\eta^2}{N}\frac{1}{N}\sum_{i,j=1}^{N}&\mathbb{E}_{\gamma_i\gamma_j}\left[\left(\operatorname{sign}(\nabla f_i(x_k)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right)\right. && (100)\\
&\left.\left(\operatorname{sign}(\nabla f_j(x_k)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_j}(x_k) < 0)\right)^\top\right] && (101)\\
= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N}&\mathbb{E}_{\gamma_i}\left[\left(\operatorname{sign}(\nabla f_i(x_k)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right)\right. && (102)\\
&\left.\left(\operatorname{sign}(\nabla f_i(x_k)) - 1 + 2\mathbb{P}(\nabla f_{\gamma_i}(x_k) < 0)\right)^\top\right] && (103)\\
= \frac{\eta^2}{N}\frac{1}{N}\sum_{i=1}^{N}&\overline{\Sigma}_i(x_k). && (104)
\end{aligned}$$

where we use independence of γ_i for all $i \in [N]$. Remembering Remark D.8, the thesis follows from Prop. B.6 and Thm. B.7. □

In all the following results, the reader will notice that all the drifts, diffusion terms, and noise assumptions are selected to guarantee that the SDE we derived for DSignSGD is indeed a 1 weak approximation for DSignSGD.

Corollary D.10. *Let us take the same assumptions of Theorem D.9, and that the stochastic gradients are $\nabla f_{\gamma_i}(x) = \nabla f(x) + \sqrt{\Sigma_i}Z_i$ such that $Z_i \sim t_\nu(0, I_d)$ does not depend on x, ν are the degrees of freedom, and scale matrices $\Sigma_i = \operatorname{diag}(\sigma_{1,i}^2, \cdots, \sigma_{d,i}^2)$. Then, the SDE of DSignSGD is*

$$dX_t = -\frac{2}{N}\sum_{i=1}^{N}\Xi_\nu\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)dt + \sqrt{\frac{\eta}{N}}\sqrt{\tilde{\Sigma}(X_t)}dW_t. \tag{105}$$

where $\Xi_\nu(x)$ is defined as $\Xi_\nu(x) := x\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\pi\nu}\Gamma\left(\frac{\nu}{2}\right)}{}_2F_1\left(\frac{1}{2}, \frac{\nu+1}{2}; \frac{3}{2}; -\frac{x^2}{\nu}\right)$, ${}_2F_1\left(a, b; c; x\right)$ is the hypergeometric function, and

$$\tilde{\Sigma}(X_t) := I_d - \frac{4}{N}\sum_{i=1}^{N}\left(\Xi_\nu\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)\right)^2. \tag{106}$$

Proof. First of all, we observe that

$$1 - 2\mathbb{P}\left(\nabla f_{\gamma_i}(x) < 0\right) = 1 - 2\mathbb{P}\left(\nabla f(x) + \Sigma_i^{\frac{1}{2}}U_i < 0\right) = 1 - 2F_\nu\left(-\Sigma_i^{-\frac{1}{2}}\nabla f(x)\right), \tag{107}$$

where $F_\nu(x)$ is the cumulative function of a t distribution with ν degrees of freedom. Remembering that

$$F_\nu(x) = \frac{1}{2} + \Xi_\nu(x), \tag{108}$$

we have that

$$1 - 2\mathbb{P}\left(\nabla f_{\gamma_i}(x) < 0\right) = 1 - 2\left(\frac{1}{2} + \Xi_\nu(-\Sigma_i^{-\frac{1}{2}}\nabla f(x))\right) = 2\Xi_\nu(\Sigma_i^{-\frac{1}{2}}\nabla f(x)). \tag{109}$$

Similarly, one can prove that $\overline{\Sigma_i}$ becomes

$$\bar{\Sigma}_i = I_d - 4\,\mathrm{diag}\left(\Xi_\nu\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)\right)^2. \tag{110}$$

□

Proposition D.11. *Under the assumptions of Corollary D.10 and signal-to-noise ratios $Y_t^i := \Sigma_i^{-\frac{1}{2}}\nabla f(X_t)$, let $\psi_\nu \in \mathbb{R}$ such that $|x| > \psi_\nu \implies 2|\Xi_\nu(x)| \sim 1$. Then, the DSignSGD has three phases:*

1. ***Phase 1:*** *If $\left|Y_t^i\right| > \psi_\nu$, the SDE coincides with the ODE of SignGD:*

$$dX_t = -sign(\nabla f(X_t))dt; \tag{111}$$

2. ***Phase 2:*** *If $1 < \left|Y_t^i\right| < \psi_\nu$:* [8]

   (a) $-m_\nu\left(\frac{1}{N}\sum_{i=1}^N \Sigma_i^{-\frac{1}{2}}\right)\nabla f(X_t) - \mathbf{q}_\nu^+ \le \frac{d\mathbb{E}[X_t]}{dt} \le -m_\nu\left(\frac{1}{N}\sum_{i=1}^N \Sigma_i^{-\frac{1}{2}}\right)\nabla f(X_t) - \mathbf{q}_\nu^-$;

   (b) $\mathbb{P}\left[\|X_t - \mathbb{E}[X_t]\|_2^2 > a\right] \le \frac{\eta}{a}\left(d - \frac{1}{N}\sum_{i=1}^N \|m_\nu Y_t^i + \mathbf{q}_\nu^-\|_2^2\right)$;

3. ***Phase 3:*** *If $\left|Y_t^i\right| < 1$ and $\ell_\nu := 2\Xi_\nu'(0)$, the SDE is*

$$dX_t = -\ell_\nu\left(\frac{1}{N}\sum_{i=1}^N \Sigma_i^{-\frac{1}{2}}\right)\nabla f(X_t)dt + \sqrt{\frac{\eta}{N}}\sqrt{I_d - \frac{\ell_\nu^2}{N}\sum_{i=1}^N \mathrm{diag}\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)^2}\,dW_t. \tag{112}$$

Proof. Exploiting the regularity of the $\Xi_\nu(x)$ function, we approximate the SDE in (105) in three different regions:

1. **Phase 1:** If $|x| > \psi_\nu$, $2\Xi_\nu(x) \sim \mathrm{sign}(x)$. Therefore, if $\left|\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right| > \psi_\nu$,

   (a) $2\Xi_\nu\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right) \sim \mathrm{sign}\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right) = \mathrm{sign}\left(\nabla f(X_t)\right)$;

   (b) $4\Xi_\nu\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)^2 \sim \mathrm{sign}\left(\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\right)^2 = (1, \dots, 1)$.

   Therefore,

$$dX_t \sim -\mathrm{sign}(\nabla f(X_t))dt; \tag{113}$$

2. **Phase 2:** If $1 < x < \psi_\nu$, we have that

$$m_\nu x + q_{\nu,1} < 2\Xi_\nu(x) < m_\nu x + q_{\nu,2}. \tag{114}$$

   Analogously, if $-\psi_\nu < x < -1$

$$m_\nu x - q_{\nu,2} < 2\Xi_\nu(x) < m_\nu x - q_{\nu,1}. \tag{115}$$

   Therefore, we have that if $1 < \left|Y_t^i\right| < \psi_\nu$, then

[8] Let m_ν and $q_{\nu,1}$ are the slope and intercept of the line secant to the graph of $2\Xi_\nu(x)$ between the points $(1, 2\Xi_\nu(1))$ and $(\psi_\nu, 2\Xi_\nu(\psi_\nu))$, while $q_{\nu,2}$ is the intercept of the line tangent to the graph of $2\Xi_\nu(x)$ and slope m_ν, $(\mathbf{q}_\nu^+)_i := \begin{cases} q_{\nu,2} & \text{if } \partial_i f(x) > 0 \\ -q_{\nu,1} & \text{if } \partial_i f(x) < 0 \end{cases}$, $(\mathbf{q}^-)_i := \begin{cases} q_{\nu,1} & \text{if } \partial_i f(x) > 0 \\ -q_{\nu,2} & \text{if } \partial_i f(x) < 0 \end{cases}$, and $\hat{q}_\nu := \max(q_{\nu,1}, q_{\nu,2})$.

(a)

$$m_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) + \mathbf{q}_\nu^- < 2\Xi_\nu \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right) < m_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) + \mathbf{q}_\nu^+. \tag{116}$$

Therefore,

$$-m_\nu \left(\frac{1}{N} \sum_{i=1}^N \Sigma_i^{-\frac{1}{2}} \right) \nabla f(X_t) - \mathbf{q}_\nu^+ \leq \frac{d\mathbb{E}\left[X_t\right]}{dt} \leq -m_\nu \left(\frac{1}{N} \sum_{i=1}^N \Sigma_i^{-\frac{1}{2}} \right) \nabla f(X_t) + \mathbf{q}_\nu^-; \tag{117}$$

(b) Similar to the above,

$$\left(m_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) + \mathbf{q}_\nu^- \right)^2 \leq 4\Xi_\nu \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right)^2 \leq \left(m_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) + \mathbf{q}_\nu^+ \right)^2.$$

Therefore,

$$\begin{aligned}
\mathbb{P}\left[\|X_t - \mathbb{E}\left[X_t\right]\|_2^2 > a \right] &\leq \mathbb{P}\left[\exists j \text{ s.t. } |X_t^j - \mathbb{E}\left[X_t^j\right]|^2 > a \right] && (118)\\
&\leq \sum_j \mathbb{P}\left[|X_t^j - \mathbb{E}\left[X_t^j\right]| > \sqrt{a} \right] \\
&\leq \frac{\eta}{a} \sum_j \left(1 - \frac{4}{N} \sum_{i=1}^N \Xi_\nu \left(\left(\Sigma_i\right)_j^{-\frac{1}{2}} \partial_j f(X_t) \right)^2 \right) && (119)\\
&< \frac{\eta}{a} \left(d - \frac{1}{N} \sum_{i=1}^N \| m_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) + \mathbf{q}_\nu^- \|_2^2 \right). && (120)
\end{aligned}$$

3. **Phase 3:** If $|x| < 1$, $2\Xi_\nu(x) \sim \ell_\nu x$ for $\ell_\nu := 2\Xi_\nu'(0)$. Therefore, if $\left| \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right| < 1$,

   (a) $2\Xi_\nu \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right) \sim \ell_\nu \Sigma_i^{-\frac{1}{2}} \nabla f(X_t)$;

   (b) $4 \left(\Xi_\nu \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right) \right)^2 \sim \ell_\nu^2 \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right)^2$.

   Therefore,

$$dX_t = -\ell_\nu \left(\frac{1}{N} \sum_{i=1}^N \Sigma_i^{-\frac{1}{2}} \right) \nabla f(X_t) dt + \sqrt{\frac{\eta}{N}} \sqrt{ I_d - \frac{\ell_\nu^2}{N} \sum_{i=1}^N \operatorname{diag} \left(\Sigma_i^{-\frac{1}{2}} \nabla f(X_t) \right)^2 } \, dW_t. \tag{121}$$

□

Theorem D.12. *Let f be μ-strongly convex, $Tr(\nabla^2 f(x)) < \mathcal{L}_\tau$, $\Sigma_i \leq \sigma_{max,i}^2$, $S_t := f(X_t) - f(X_*)$, and $\sigma_{\mathcal{H},j}$ be the harmonic mean of $\{(\sigma_{max,i})^j\}$. Then, if all agents are in*

1. *Phase 1, the loss will reach 0 before $t_* = 2\sqrt{\frac{S_0}{\mu}}$ because $S_t \leq \frac{1}{4}\left(\sqrt{\mu}t - 2\sqrt{S_0}\right)^2$;*
2. *Phase 2, $\mathbb{E}[S_t] \leq S_0 e^{-2\mu\Delta t} + \frac{\eta(\mathcal{L}_\tau - \mu d\hat{q}^2)}{2N} \frac{1}{2\mu\Delta} \left(1 - e^{-2\mu\Delta t}\right)$ with $\Delta := m_\nu \sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu m_\nu^2}{2N} \sigma_{\mathcal{H},2}^{-1}$;*
3. *Phase 3, $\mathbb{E}[S_t] \leq S_0 e^{-2\mu\Delta t} + \frac{\eta\mathcal{L}_\tau}{2N} \frac{1}{2\mu\Delta} \left(1 - e^{-2\mu\Delta t}\right)$ with $\Delta := \ell_\nu \sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu\ell_\nu^2}{2N} \sigma_{\mathcal{H},2}^{-1}$.*

Proof. Let us prove the above phase by phase:

For Phase 1,

$$d(f(X_t) - f(X_*)) = -\nabla f(X_t) \operatorname{sign}(\nabla f(X_t)) dt = -\|\nabla f(X_t)\|_1 dt \leq -\|\nabla f(X_t)\|_2 dt. \tag{122}$$

Since f is μ-PL, we have that $-\|\nabla f(X_t)\|_2^2 < -2\mu(f(X_t) - f(X_*))$, which implies that

$$f(X_t) - f(X_*) \leq \frac{1}{4} \left(\sqrt{\mu}t - 2\sqrt{f(X_0) - f(X_*)} \right)^2, \tag{123}$$

meaning that the dynamics will stop before $t_* = 2\sqrt{\frac{f(X_0) - f(X_*)}{\mu}}$;

For Phase 2, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) = -\frac{m_\nu}{N}\sum_{i=1}^{N}\nabla f(X_t)^\top \Sigma_i^{-\frac{1}{2}}\nabla f(X_t)dt - \nabla f(X_t)^\top \mathbf{q}_\nu^- dt + \frac{\eta\mathcal{L}_\tau}{2N}dt \tag{124}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{\eta\mu}{2N}\frac{1}{N}\sum_{i=1}^{N}\|m_\nu \Sigma_i^{-\frac{1}{2}}\nabla f(X_t) + \mathbf{q}_\nu\|_2^2 dt \tag{125}$$

$$\leq -m_\nu\sigma_{\mathcal{H},1}^{-1}\|\nabla f(X_t)\|_2^2 dt - \hat{q}\|\nabla f(X_t)\|_1 dt + \frac{\eta\mathcal{L}_\tau}{2N}dt \tag{126}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{\eta\mu d\hat{q}^2}{2N}dt - \frac{\eta m_\nu^2\mu}{2N}\sigma_{\mathcal{H},2}^{-1}\|\nabla f(X_t)\|_2^2 dt - \frac{\eta\mu m_\nu\sigma_{\mathcal{H},1}^{-1}\hat{q}}{N}\|\nabla f(X_t)\|_1 dt \tag{127}$$

$$\leq -\left(m_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu m_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}\right)\|\nabla f(X_t)\|_2^2 dt + \frac{\eta(\mathcal{L}_\tau - \mu d\hat{q}^2)}{2N}dt + \mathcal{O}(\text{Noise}) \tag{128}$$

$$\leq -2\mu\left(m_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu m_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}\right)(f(X_t) - f(X_*))dt + \frac{\eta(\mathcal{L}_\tau - \mu d\hat{q}^2)}{2N}dt + \mathcal{O}(\text{Noise}) \tag{129}$$

meaning that

$$\mathbb{E}[f(X_t) - f(X_*)] \leq (f(X_0) - f(X_*))e^{-2\mu\Delta t} + \frac{\eta(\mathcal{L}_\tau - \mu d\hat{q}^2)}{2N}\frac{1}{2\mu\Delta}\left(1 - e^{-2\mu\Delta t}\right) \tag{130}$$

with $\Delta := m_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu m_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}$, $\sigma_{\mathcal{H},j}$ is the harmonic mean of $\{(\sigma_{\max,i})^j\}$.

For Phase 3, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) = -\frac{\ell_\nu}{N}\sum_{i=1}^{N}\nabla f(X_t)^\top \Sigma_i^{-\frac{1}{2}}\nabla f(X_t)dt + \frac{\eta\mathcal{L}_\tau}{2N}dt \tag{131}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{\eta}{2N}\frac{\ell_\nu^2}{N}\sum_{i=1}^{N}\|\nabla^2 f(X_t)\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\|_2^2 dt \tag{132}$$

$$\leq -\ell_\nu\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{\sigma_{\max,i}}\right)\|\nabla f(X_t)\|_2^2 dt + \frac{\eta\mathcal{L}_\tau}{2N}dt \tag{133}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{\eta\mu\ell_\nu^2}{2N}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{1}{\sigma_{\max,i}^2}\right)\|\nabla f(X_t)\|_2^2 dt \tag{134}$$

$$= -\ell_\nu\sigma_{\mathcal{H},1}^{-1}\|\nabla f(X_t)\|_2^2 dt + \frac{\eta\mathcal{L}_\tau}{2N}dt + \mathcal{O}(\text{Noise}) - \frac{\eta\mu\ell_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}\|\nabla f(X_t)\|_2^2 dt \tag{135}$$

$$\leq -\left(\ell_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu\ell_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}\right)\|\nabla f(X_t)\|_2^2 dt + \frac{\eta\mathcal{L}_\tau}{2N}dt + \mathcal{O}(\text{Noise}) \tag{136}$$

$$\leq -2\mu\left(\ell_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu\ell_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}\right)(f(X_t) - f(X_*))dt + \frac{\eta\mathcal{L}_\tau}{2N}dt + \mathcal{O}(\text{Noise}) \tag{137}$$

meaning that

$$\mathbb{E}[f(X_t) - f(X_*)] \leq (f(X_0) - f(X_*))e^{-2\mu\Delta t} + \frac{\eta\mathcal{L}_\tau}{2N}\frac{1}{2\mu\Delta}\left(1 - e^{-2\mu\Delta t}\right) \tag{138}$$

with $\Delta := \ell_\nu\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\mu\ell_\nu^2}{2N}\sigma_{\mathcal{H},2}^{-1}$, $\sigma_{\mathcal{H},j}$ is the harmonic mean of $\{(\sigma_{\max,i})^j\}$, and $\ell_\nu := 2\Xi_\nu^{'}(0)$. □

Remark: If not all the agents are in the same Phase, we can upper bound the dynamics of df_t with the case where they are all in the third Phase, which is that of weakest descent.

Theorem D.13. *Let f be μ-PL, L-Smooth, $\Sigma_i \leq \sigma_{max,i}^2$, $S_t := f(X_t) - f(X_*)$, and $\sigma_{\mathcal{H},j}$ be the harmonic mean of $\{(\sigma_{max,i})^j\}$. Then, if all agents are in*

1. *Phase 1, the loss will reach 0 before $t_* = 2\sqrt{\frac{S_0}{\mu}}$ because $S_t \leq \frac{1}{4}\left(\sqrt{\mu}t - 2\sqrt{S_0}\right)^2$;*
2. *Phase 2, $\mathbb{E}[S_t] \leq S_0 e^{-2\mu\Delta t} + \frac{\eta L d}{4\mu\Delta N}\left(1 - e^{-2\mu\Delta t}\right)$ with $\Delta := m_\nu\sigma_{\mathcal{H},1}^{-1}$;*
3. *Phase 3, $\mathbb{E}[S_t] \leq S_0 e^{-2\mu\Delta t} + \frac{\eta L d}{4\mu\Delta N}\left(1 - e^{-2\mu\Delta t}\right)$ with $\Delta := \ell_\nu\sigma_{\mathcal{H},1}^{-1}$;*

Proof. Let us prove the above phase by phase:

For Phase 1,

$$d(f(X_t) - f(X_*)) = -\nabla f(X_t)\text{sign}(\nabla f(X_t))dt = -\|\nabla f(X_t)\|_1 dt \leq -\|\nabla f(X_t)\|_2 dt. \tag{139}$$

Since f is μ-PL, we have that $-\|\nabla f(X_t)\|_2^2 < -2\mu(f(X_t) - f(X_*))$, which implies that

$$f(X_t) - f(X_*) \leq \frac{1}{4}\left(\sqrt{\mu}t - 2\sqrt{f(X_0) - f(X_*)}\right)^2, \tag{140}$$

meaning that the dynamics will stop before $t_* = 2\sqrt{\frac{f(X_0)-f(X_*)}{\mu}}$;

For Phase 2, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) = -\frac{m_\nu}{N}\sum_{i=1}^{N}\nabla f(X_t)^\top \Sigma_i^{-\frac{1}{2}}\nabla f(X_t)dt - \nabla f(X_t)^\top \mathbf{q}_\nu^- dt + \frac{\eta L d}{2N}dt \tag{141}$$

$$\leq -2\mu m_\nu \sigma_{\mathcal{H},1}^{-1}(f(X_t) - f(X_*))dt + \frac{\eta L d}{2N}dt + \mathcal{O}(\text{Noise}) \tag{142}$$

which implies the thesis.

For Phase 3, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) = -\frac{\ell_\nu}{N}\sum_{i=1}^{N}\nabla f(X_t)^\top \Sigma_i^{-\frac{1}{2}}\nabla f(X_t)dt - \nabla f(X_t)^\top \mathbf{q}_\nu^- dt + \frac{\eta L d}{2N}dt \tag{143}$$

$$\leq -2\mu \ell_\nu \sigma_{\mathcal{H},1}^{-1}(f(X_t) - f(X_*))dt + \frac{\eta L d}{2N}dt + \mathcal{O}(\text{Noise}) \tag{144}$$

which implies the thesis. □

Theorem D.14. *If f is L-smooth, we use a learning rate scheduler η_t such that $\phi_t^i = \int_0^t (\eta_s)^i ds$, $\phi_t^1 \overset{t\to\infty}{\to} \infty$, $\frac{\phi_t^2}{\phi_t^1} \overset{t\to\infty}{\to} 0$, and $\Sigma_i \leq \sigma_{max,i}^2$.*

1. *In Phase 1, $\|\nabla f(X_{\tilde{t}^1})\|_1 \leq \frac{f(X_0)-f(X_*)}{\phi_t^1} \overset{t\to\infty}{\to} 0$;*
2. *In Phase 2,*

$$m_\nu \mathbb{E}\|\nabla f(X_{\tilde{t}^{(1,2)}})\|_2^2 + \hat{q}\sigma_{\mathcal{H},1}\mathbb{E}\|\nabla f(X_{\tilde{t}^{(2,2)}})\|_1 \leq \frac{\sigma_{\mathcal{H},1}}{\phi_t^1}\left(f(X_0) - f(X_*) + \frac{\eta L d \phi_t^2}{2N}\right) \overset{t\to\infty}{\to} 0; \tag{145}$$

3. *In Phase 3,*

$$\ell_\nu \mathbb{E}\|\nabla f(X_{\tilde{t}^3})\|_2^2 \leq \frac{\sigma_{\mathcal{H},1}}{\phi_t^1}\left(f(X_0) - f(X_*) + \frac{\eta L d \phi_t^2}{2N}\right) \overset{t\to\infty}{\to} 0. \tag{146}$$

Above, $\tilde{t}^1$, $\tilde{t}^{(1,2)}$, $\tilde{t}^{(2,2)}$, and $\tilde{t}^3$ are random times with distribution $\frac{\eta_t}{\phi_t^1}$.

Proof. Let us prove the above phase by phase:

For Phase 1,

$$d(f(X_t) - f(X_*)) = -\eta_t \nabla f(X_t)\text{sign}(\nabla f(X_t))dt = -\eta_t \|\nabla f(X_t)\|_1 dt \tag{147}$$

$$= -\phi_t^1 \frac{\eta_t \|\nabla f(X_t)\|_1}{\phi_t^1}dt \tag{148}$$

Therefore, by integrating over time and using the law of the unconscious statistician

$$\|\nabla f(X_{\tilde{t}^1})\|_1 \leq \frac{f(X_0) - f(X_*)}{\phi_t^1} \overset{t\to\infty}{\to} 0; \tag{149}$$

where $\tilde{t}^1$ is a random time with distribution $\frac{\eta_t}{\phi_t^1}$; □

For Phase 2, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) \leq -\eta_t m_\nu \sigma_{\mathcal{H},1}^{-1} \|\nabla f(X_t)\|_2^2 dt - \eta_t \hat{q} \|\nabla f(X_t)\|_1 dt + \eta_t^2 \frac{\eta L d}{2N} dt + \mathcal{O}(\text{Noise}) \tag{150}$$

Therefore, by integrating over time and using the law of the unconscious statistician we have

$$m_\nu \mathbb{E}\|\nabla f\left(X_{\tilde{t}^{(1,2)}}\right)\|_2^2 + \hat{q}\sigma_{\mathcal{H},1}\mathbb{E}\|\nabla f\left(X_{\tilde{t}^{(2,2)}}\right)\|_1 \leq \frac{\sigma_{\mathcal{H},1}}{\phi_t^1}\left(f(X_0) - f(X_*) + \frac{\eta L d \phi_t^2}{2N}\right) \overset{t\to\infty}{\to} 0, \tag{151}$$

where $\tilde{t}^{(1,2)}$, $\tilde{t}^{(2,2)}$, and $\tilde{t}^3$ are random times with distribution $\frac{\eta_t}{\phi_t^1}$;

For Phase 3, using Ito on f, we have that

$$d(f(X_t) - f(X_*)) \leq -\eta_t \ell_\nu \sigma_{\mathcal{H},1}^{-1} \|\nabla f(X_t)\|_2^2 dt + \eta_t^2 \frac{\eta L d}{2N} dt + \mathcal{O}(\text{Noise}) \tag{152}$$

Therefore, by integrating over time and using the law of the unconscious statistician we have

$$\ell_\nu \mathbb{E}\|\nabla f\left(X_{\tilde{t}^3}\right)\|_2^2 \leq \frac{\sigma_{\mathcal{H},1}}{\phi_t^1}\left(f(X_0) - f(X_*) + \frac{\eta L d \phi_t^2}{2N}\right) \overset{t\to\infty}{\to} 0, \tag{153}$$

where $\tilde{t}^3$ is a random time with distribution $\frac{\eta_t}{\phi_t^1}$.

D.1 Scaling Rules

Proposition D.15. *Let the batch size be δB, learning rate $\kappa\eta$, and αN agents. Let $K_1 := \ell_\nu \sqrt{B}\sigma_{\mathcal{H},1}^{-1}$ and $K_2 := \frac{\eta \ell_\nu^2 B \mu \sigma_{\mathcal{H},2}^{-1}}{2N}$. The scaling rules (involving only two parameters at the time) to preserve the performance independently of δ, κ, and α, are: Finally, we observe that if $\frac{K_1}{K_2} \sim 0$, for example when $N \gg 1$, then these rules can be summarized as $\frac{\kappa}{\alpha\sqrt{\delta}} = 1$, which*

Scaling Rule	Implication
$\alpha = \frac{1}{\sqrt{\delta}} + \frac{K_2}{K_1}\left(\frac{1}{\sqrt{\delta}} - \sqrt{\delta}\right)$	BS $\downarrow \implies$ Agents $\uparrow$
$\alpha = \kappa$	LR $\uparrow \implies$ Agents $\uparrow$
$\kappa = \frac{\sqrt{\delta}}{1 + \frac{K_1}{K_2}(1-\delta)}$	BS $\uparrow \implies$ Agents $\uparrow$

Table 3: Summary of Trade-offs Between Parameters (LR = Learning Rate and BS = Batch Size).

recover the Scaling Rules of Adam and RMSprop as well as allow for the enhanced design flexibility of the distributed setting.

Proof. Let us focus on Phase 3, which is when the dynamics reaches stability. Using Ito's Proposition on f we have that

$$d(f(X_t) - f(X_*)) = -\frac{\kappa}{\alpha N}\sum_{i=1}^{\alpha N} \ell_\nu \sqrt{\delta B}\nabla f(X_t)^\top \Sigma_i^{-\frac{1}{2}} \nabla f(X_t) dt + \frac{\eta\kappa^2 \mathcal{L}_\tau}{2\alpha N} dt \tag{154}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{\eta\kappa^2}{2\alpha N}\frac{\ell_\nu^2 B\delta}{\alpha N}\sum_{i=1}^{\alpha N}\|\nabla^2 f(X_t)\Sigma_i^{-\frac{1}{2}}\nabla f(X_t)\|_2^2 dt \tag{155}$$

$$\leq -(2\mu\kappa\ell_\nu\sqrt{\delta B})\left(\frac{1}{\alpha N}\sum_{i=1}^{\alpha N}\frac{1}{\sigma_{\max,i}}\right)(f(X_t) - f(X_*))dt + \frac{\eta\kappa^2\mathcal{L}_\tau}{2\alpha N}dt \tag{156}$$

$$+ \mathcal{O}(\text{Noise}) - \frac{2\mu^2\eta\kappa^2}{2\alpha N}\ell_\nu^2 B\delta\left(\frac{1}{\alpha N}\sum_{i=1}^{\alpha N}\frac{1}{\sigma_{\max,i}^2}\right)(f(X_t) - f(X_*))dt, \tag{157}$$

meaning that

$$\mathbb{E}[f(X_t) - f(X_*)] \leq (f(X_0) - f(X_*))e^{-2\mu\Delta t} + \frac{\eta\kappa^2}{2\alpha N}\frac{\mathcal{L}_\tau}{2\mu\Delta}\left(1 - e^{-2\mu\Delta t}\right) \tag{158}$$

with $\Delta := \left(\ell_\nu\kappa\sqrt{\delta B}\sigma_{\mathcal{H},1}^{-1} + \frac{\eta\kappa^2}{2\alpha N}\ell_\nu^2\delta B\mu\sigma_{\mathcal{H},2}^{-1}\right)$.

The asymptotic limit is thus:

$$\frac{\eta\mathcal{L}_\tau}{4\mu N}\frac{\kappa}{\alpha\sqrt{\delta}}\frac{1}{\ell_\nu\sqrt{B}\sigma_{\mathcal{H},1}^{-1}+\frac{\eta\ell_\nu^2 B\mu\sigma_{\mathcal{H},2}^{-1}}{2N}\frac{\kappa\sqrt{\delta}}{\alpha}}. \tag{159}$$

To maintain the performance of DSignSGD independently of α, κ, and δ, we need to solve the following equation:

$$\frac{\eta\mathcal{L}_\tau}{4\mu N}\frac{\kappa}{\alpha\sqrt{\delta}}\frac{1}{\ell_\nu\sqrt{B}\sigma_{\mathcal{H},1}^{-1}+\frac{\eta\ell_\nu^2 B\mu\sigma_{\mathcal{H},2}^{-1}}{2N}\frac{\kappa\sqrt{\delta}}{\alpha}}=\frac{\eta\mathcal{L}_\tau}{4\mu N}\frac{1}{\ell_\nu\sqrt{B}\sigma_{\mathcal{H},1}^{-1}+\frac{\eta\ell_\nu^2 B\mu\sigma_{\mathcal{H},2}^{-1}}{2N}}. \tag{160}$$

To provide easily interpretable and actionable scaling rules, we fix one of the three parameters to 1 and find the relationship between the others. With simple math, the thesis follows. □

D.2 Stationary Distribution

Proposition D.16. *Let* $H=\operatorname{diag}(\lambda_1,\dots,\lambda_d)$, $M_t:=e^{-2\left(\ell_\nu\Sigma_{\mathcal{H},1}H+\frac{\eta}{2N}\ell_\nu^2\Sigma_{\mathcal{H},2}H^2\right)t}$ *where* $\Sigma_{\mathcal{H},1}=\frac{1}{N}\sum_{i=1}^N\Sigma_i^{-\frac{1}{2}}$, *and* $\Sigma_{\mathcal{H},2}=\frac{1}{N}\sum_{i=1}^N\Sigma_i^{-1}$. *Then,*

1. $\mathbb{E}[X_t]=e^{-\ell_\nu\Sigma_{\mathcal{H},1}Ht}X_0\overset{t\to\infty}{\to}0$;
2. $Cov[X_t]=\left(M_t-e^{-2\ell_\nu\Sigma_{\mathcal{H},1}Ht}\right)X_0^2+\frac{\eta}{2N}\left(\ell_\nu I_d+\frac{\eta}{2N}\ell_\nu^2\Sigma_{\mathcal{H},2}\Sigma_{\mathcal{H},1}^{-1}H\right)^{-1}H^{-1}\Sigma_{\mathcal{H},1}^{-1}\left(I_d-M_t\right)$, *which as* $t\to\infty$ *converges to* $\frac{\eta}{2N}\left(\ell_\nu I_d+\frac{\eta}{2N}\ell_\nu^2\Sigma_{\mathcal{H},2}\Sigma_{\mathcal{H},1}^{-1}H\right)^{-1}H^{-1}\Sigma_{\mathcal{H},1}^{-1}$.

Proof. The proof mimics that of Prop. C.13. □

E Additional related works

In this section, we list some papers that derived or used SDEs to model optimizers. In particular, we focus on the aspect of empirically verifying the validity of such SDEs in the sense that they indeed track the respective optimizers. We divide these into three categories: Those that did not carry out any type of validation, those that did it on simple landscapes (quadratic functions et similia), and those that did small experiments on neural networks.

None of the following papers carried out any experimental validation of the approximating power of the SDEs they derived. Many of them did not even validate the insights derived from the SDEs: (75, 49, 11, 123, 25, 82, 109, 65, 8, 101, 68, 110, 10, 17, 63, 119, 105, 72, 39, 26, 81).

The following ones carried out validation experiments on artificial landscapes, e.g. quadratic or quartic function, or easy regression tasks: (69, 70, 122, 4, 35, 43, 103, 5).

The following papers carried out some experiments which include neural networks: (88, 20). In particular, they both simulate the SDEs with a numerical integrator and compare them with the respective optimizers: The first validates the SDE on a shallow MLP while the second does so on a shallow and a deep MLP. We also verify our SDEs on simple landscapes as well as on an MPL (on Breast Cancer) and, importantly, we verify our insights on ViTs (on MNIST) and ResNets (on CIFAR-10).

It would be great to extend the theoretical results to a more practical class of structural non-convex problems, e.g., under quasar convexity (45), α-β-condition (51), or Aiming condition (74).

F EXPERIMENTS

In this section, we provide the modeling choices and instructions to replicate our experiments. The code is implemented in Python 3 (106) mainly using Numpy (46), scikit-learn (89), and JAX (15).

F.1 SDE validation (Figure 1)

In this subsection, we describe the experiments we run to produce Figure 1: The trajectories of the SDEs match those of the respective algorithms on average. Additionally, the SDEs and the algorithms move at the same speed.

DSGD - Rosenbrock This paragraph refers to the *left* of Figure 1. The loss function is the Rosenbrock function with parameters $a=1$ and $b=100$. We run DSGD for 10000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z\sim\mathcal{N}(0,\sigma^2 I_d)$ where $\sigma=100$. The learning rate is $\eta=0.001$ and $N=10$. Similarly, we integrate the DSGD SDE (Thm. 3.2) with Euler-Maruyama (Algorithm 1) with $\Delta t=\eta$. Results are averaged over 5000. We plot the averaged trajectories and observe that they overlap to a great degree of agreement.

DCSGD - Embedded Saddle This paragraph refers to the *center-left* of Figure 1. We optimize the function $f(x) = \frac{x^\top Hx}{2} + \frac{1}{4}\lambda\sum_{i=1}^{2} x_i^4 - \frac{\xi}{3}\sum_{i=1}^{2} x_i^3$ where $H = \text{diag}(1, -2)$, $\lambda = 1$, and $\xi = 1$. We run DCSGD with Rand-k as $k = 1$ for 1000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 10$. The learning rate is $\eta = 0.1$ and $N = 10$. Similarly, we integrate the DCSGD SDE (Thm. 3.6) with Euler-Maruyama (Algorithm 1) with $\Delta t = \eta$. Results are averaged over 5000. We plot the averaged trajectories and observe that they overlap to a great degree of agreement.

DSignSGD - Convex Quadratic Function This paragraph refers to the *center-right* of Figure 1. We optimize the function $f(x) = \frac{x^\top Hx}{2}$ where $H = \text{diag}(5, 5)$. We run DSignSGD for 3000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \text{diag}(\sigma_i^2))$ where $\sigma_i = 0.01 * (1 + i)$, the learning rate is $\eta = 0.001$ and $N = 10$. Similarly, we integrate the DSignSGD SDE (Thm. 3.10) with Euler-Maruyama (Algorithm 1) with $\Delta t = \eta$. Results are averaged over 5000. We plot the averaged trajectories and observe that they overlap to a great degree of agreement.

DNN on Breast Cancer Dataset (33) This paragraph refers to the *right* of Figure 1. The DNN has 10 dense layers with 20 neurons each activated with a ReLu. We minimize the binary cross-entropy loss. We run DCSGD with Rand-k and $k = 1000$, $d = 1502$, and for 30000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.0001$. The learning rate is $\eta = 0.1$ and $N = 3$. Similarly, we integrate the DCSGD SDE (Thm. 3.6) with Euler-Maruyama (Algorithm 1) with $\Delta t = \eta$. Results are averaged over 3 runs and the shaded areas are the average $\pm$ the standard deviation.

F.2 DCSGD - Scaling Rules (Figure 2)

Transformer on MNIST (29) This paragraph refers to the *left* of Figure 2. The Architecture is a scaled-down version of (32), where the hyperparameters are *patch size*=28, *out features*=10, *width*=48, *depth*=3, *num heads*=6, and *dim ffn*=192. We minimize the cross-entropy loss. In this experiment, we run DSGD with some hyperparameters (η, B, N) for 1000 epochs. Then, we need to verify the scaling rules in Prop. 3.9, meaning that we run DCSGD with hyperparameters that follow the rules reported there and confirm that they indeed recover the performance of DSGD. Then, we also run DCSGD with combinations of hyperparameters that do not do so and indeed they do not recover the performance of DSGD. In all our experiments, we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.01$ which corresponds to $B = 1$. The learning rate is $\eta = 0.01$ and the number of agents is $N = 3$. DCSGD(η, B, ω, N) is with $\omega = 1$ and indeed does not perform as DSGD(η, B, N). DCSGD$(\eta, B, \omega, (1+\omega)N)$ almost recovers the performance of DSGD(η, B, N). The same with DCSGD$(\eta, B, \beta\omega, (1+\beta\omega)N)$, DCSGD$(\eta, (1+\omega)B, \omega, N)$, and DCSGD$(\eta, (1+\beta\omega)B, \beta\omega, N)$ for $\beta = 2$. On the contrary, neither DCSGD$(\kappa\eta, B, \beta\omega, (1+\beta\omega)N)$ for $\kappa = 3$ nor DCSGD$(\eta, \delta B, \omega, N)$ for $\delta = 1/3$ do so because they do not satisfy our scaling rules. See Figure 6 for a boxplot comparing the errors at the last iterate: Clearly, those hyperparameter combinations that do not follow our prescriptions behave much differently DSGD than those that do follow our rules. Results are averaged over 50 runs.

ResNet on CIFAR-10 (62) This paragraph refers to the *left* of Figure 2. The ResNet has a $(3, 3, 32)$ convolutional layer with stride 1, followed by a ReLu activation, a second $(3, 3, 32)$ convolutional layer with stride 1, followed by a residual connection from the first convolutional layer, then a $(2, 2)$ max pool layer with stride $(2, 2)$. Then the activations are flattened and passed through a dense layer that compresses them into 128 dimensions, a final ReLu activation, and a final dense layer into the output dimension 10. The output finally goes through a softmax as we minimize the cross-entropy loss. In this experiment, we run DSGD with some hyperparameters (η, B, N). Then, we need to verify the scaling rules in Prop. 3.9, meaning that we run DCSGD with hyperparameters that follow the rules reported there and confirm that they indeed recover the performance of DSGD. Then, we also run it with a combination that does not do so and indeed it does not recover the performance of DSGD. In all our experiments, we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.01$ which corresponds to $B = 1$. The learning rate is $\eta = 0.01$ and the number of agents is $N = 3$. DCSGD(η, B, ω, N) is with $\omega = 1$ and indeed does not perform as DSGD(η, B, N). DCSGD$(\eta, B, \omega, (1+\omega)N)$ almost recovers the performance of DSGD(η, B, N). The same with DCSGD$(\eta, B, \beta\omega, (1+\beta\omega)N)$, DCSGD$(\eta, (1+\omega)B, \omega, N)$, and DCSGD$(\eta, (1+\beta\omega)B, \beta\omega, N)$ for $\beta = 2$. On the contrary, neither DCSGD$(\kappa\eta, B, \beta\omega, (1+\beta\omega)N)$ for $\kappa = 3$ nor DCSGD$(\eta, \delta B, \omega, N)$ for $\delta = 1/3$ do so because they do not satisfy our scaling rules. See Figure 6 for a boxplot comparing the errors at the last iterate: Clearly, those hyperparameter combinations that do not follow our prescriptions behave much differently DSGD than those that do follow our rules. Results are averaged over 10 runs.

F.3 DSignSGD - Bound and Linear Speedup (Figure 3)

Bound - Left In this paragraph, we describe how we validated the existence of the phases of DSignSGD as predicted in Thm. 3.12. We run DSignSGD with $\eta = 0.001$ for 800 epochs, $N = 12$ as we optimize function is $f(x) = \frac{x^\top Hx}{2}$ for $H = \text{diag}(2)$, and inject Gaussian noise with covariance matrix $\Sigma = \sigma^2 I_d$ where $\sigma = 0.1$ on the full gradient. We plot the bounds as per Thm. 3.12 and confirm that they indeed match or bound the dynamics of the loss as prescribed. Results are averaged over 100 runs.

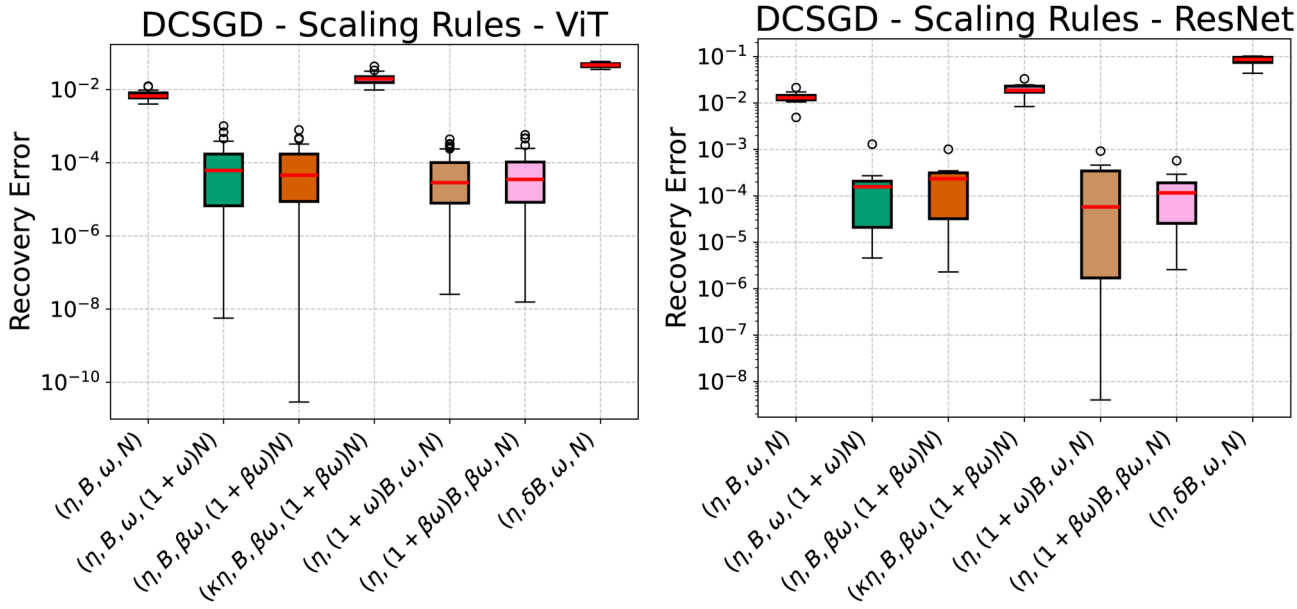

Figure 6: Box plot of the error between the last iterate of DSGD base run and the runs of DCSGD with the different combinations of hyperparameters: Those runs that follow our Scaling Rules achieve a much smaller error than those that do not.

Linear Speedup - Right In this paragraph, we describe how we validated the linear speedup of DSignSGD on the ViT described above. We run DSignSGD as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 1$, $\eta = 0.01$ and $N \in \{1, 2, 4, 8, 16\}$. Results are averaged over 3 runs.

F.4 DSignSGD - Scaling Rules (Figure 4)

Transformer on MNIST (29) This paragraph refers to the *left* of Figure 4. The ViT is the same as described above. In this experiment, we run DSignSGD with some hyperparameters (η, B, N). Then, we need to verify the scaling rules in Prop. 3.14, meaning that we run DSignSGD with hyperparameters that follow the rules reported there and confirm that they indeed preserve the performance. Then, we also run it with a combination that does not do so and indeed it does not preserve them. In all our experiments, we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.2$ which corresponds to $B = 1$. The learning rate is $\eta = 0.01$ for 1000 epochs and the number of agents is $N = 4$. Since they follow our scaling rules, DSignSGD$(\kappa\eta, \kappa^2 B, N)$, DSignSGD$(\kappa\eta, B, \kappa N)$, and DSignSGD$(\eta, \kappa^2 B, N/\kappa)$ with $\kappa = 2$ indeed preserve the performance of DSignSGD(η, B, N), while DSignSGD$(\kappa\eta, \kappa^2 B, N/\kappa)$ does not. See Figure 7 for a boxplot comparing the errors at the last iterate: Clearly, those hyperparameter combinations that do not follow our prescriptions behave much differently than the base run than those that do follow our rules. Results are averaged over 5 runs.

ResNet on CIFAR-10 (62) This paragraph refers to the *left* of Figure 4. The ResNet is the same as we described above. In this experiment, we run DSignSGD with some hyperparameters (η, B, N). Then, we need to verify the scaling rules in Prop. 3.14, meaning that we run DSignSGD with hyperparameters that follow the rules reported there and confirm that they indeed preserve the performance. Then, we also run it with a combination that does not do so and indeed it does not preserve them. In all our experiments, we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 1$ which corresponds to $B = 1$. The learning rate is $\eta = 0.01$ for 2000 epochs and the number of agents is $N = 4$. Since they follow our scaling rules, DSignSGD$(\kappa\eta, \kappa^2 B, N)$, DSignSGD$(\kappa\eta, B, \kappa N)$, and DSignSGD$(\eta, \kappa^2 B, N/\kappa)$ with $\kappa = 2$ indeed preserve the performance of DSignSGD(η, B, N), while DSignSGD$(\kappa\eta, \kappa^2 B, N/\kappa)$ does not. See Figure 7 for a boxplot comparing the errors at the last iterate: Clearly, those hyperparameter combinations that do not follow our prescriptions behave much differently than the base run than those that do follow our rules. Results are averaged over 10 runs.

F.5 Heavy Tailed and Large Noise (Figure 5)

The ViT is the same as above for each sub-figure.

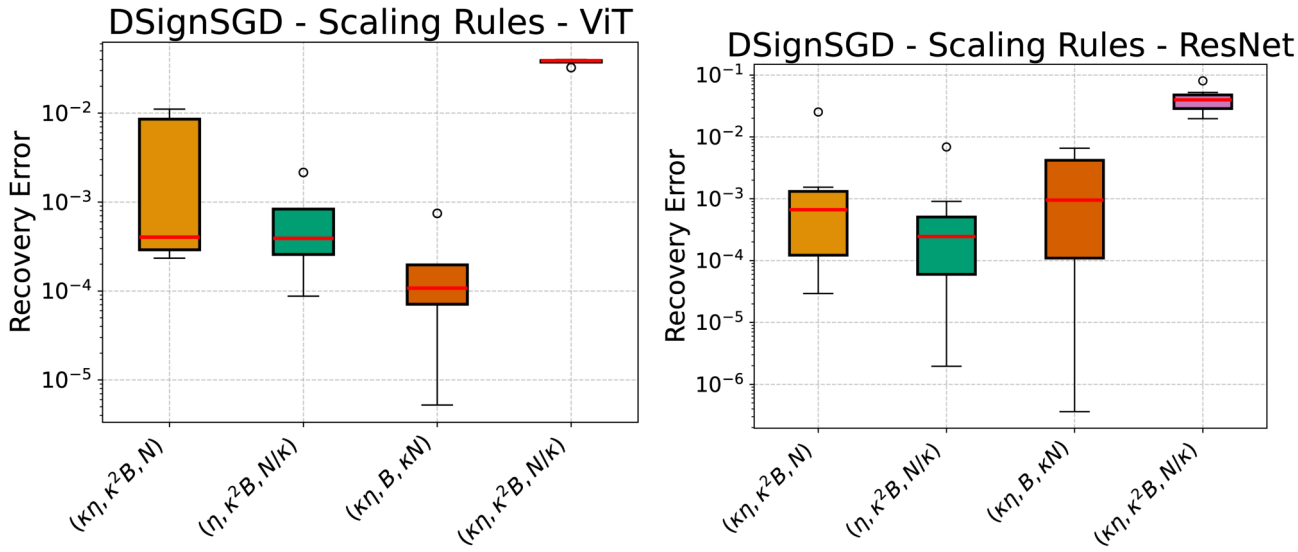

Figure 7: Box plot of the error between the last iterate of DSignSGD's base run and the runs with the different combinations of hyperparameters: Those runs that follow our Scaling Rules achieve a much smaller error than those that do not.

DCSGD - Heavy-Tailed Noise We train the ViT with DCSGD with Rand-k where $k = 100000$ out of $d = 133930$, $\eta = 0.01$ for 1000 epochs, and as we inject noise distributed as a Student's t with scale $\Sigma = \sigma^2 I_d$ and $\nu \in \{1, 2, 3, 8, 64, \infty\}$, and $N = 3$. Even if the scale is small ($\sigma^2 = 10^{-8}$): 1) When $\nu = 1$, the optimizer diverges; 2) When $\nu = 2$, the loss is non-stationary; 3) The larger ν, the more stable and optimal the loss. This confirms that indeed DCSGD cannot handle Heavy-Tailed noise. Results are averaged over 3 runs.

DCSGD - Large Noise We train the ViT with DCSGD with Rand-k where $k = 100000$ out of $d = 133930$, $\eta = 0.01$ for 5000 epochs, and as we inject noise distributed as a Gaussian with covariance matrix $\sigma^2 \in \{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 10^{0}\}$, and $N = 3$. As the variance increases, the optimizer diverges more and more: This confirms that indeed DCSGD cannot handle large noise as its loss level scales quadratically in the noise level. Results are averaged over 3 runs.

DSignSGD - Heavy-Tailed Noise We train the ViT with DSignSGD as we inject noise distributed as a Student's t with scale $\Sigma = \sigma^2 I_d$ and $\nu \in \{1, 2, 3, 8, 64, \infty\}$, and $N = 3$, $\eta = 0.01$ for 1000 epochs. Even if the scale is large ($\sigma^2 = 1$) and the noise is of unbounded expected value, DSignSGD never diverges. Of course, fatter tails imply larger loss: This confirms that indeed DSignSGD can handle Heavy-Tailed noise. Results are averaged over 3 runs.

DSignSGD - Large Noise We train the ViT with DSignSGD as we inject noise distributed as a Gaussian with covariance matrix $\Sigma = \sigma^2 I_d$ and $\sigma^2 \in \{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 10^{0}\}$, and $N = 3$, $\eta = 0.01$ for 8000 epochs. As the variance increases, the optimizer never diverges: This confirms that indeed DSignSGD can handle large noise as its loss level scales linearly in the noise level. Results are averaged over 3 runs.

F.6 DCSGD - Divergence, Bound, and Linear Speedup (Figure 8)

Divergence We optimize the function $f(x) = \frac{x^\top H x}{2}$ where $H = \mathrm{diag}(100 I_{128})$. We run DCSGD with Rand-k for 25 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.1$. The learning rate is $\eta = 0.01$. As we decrease $k \in \{128, 64, 32, 16, 8, 4, 2, 1\}$, we see that the convergence slows down and reaches a larger and larger asymptotic loss value, up to diverging. Results are averaged over 5000 runs.

Linear Speedup In this paragraph, we describe how we validated the linear speedup of DCSGD on the same ViT as above. We run DCSGD with Rand-k with $k = 100000$ as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.01$, $\eta = 0.01$ and $N \in \{1, 2, 4, 8, 16\}$. Averaged over 3 runs.

Bound In this paragraph, we describe how we validated the bound DCSGD as predicted in Thm. 3.7. We run DCSGD with Rand-k for 2000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.1$, $\eta = 0.01$, $k = 2$ and $N = 12$ as we optimize function is $f(x) = \frac{x^\top H x}{2}$ for $H = I_{100}$. We plot the bounds as per

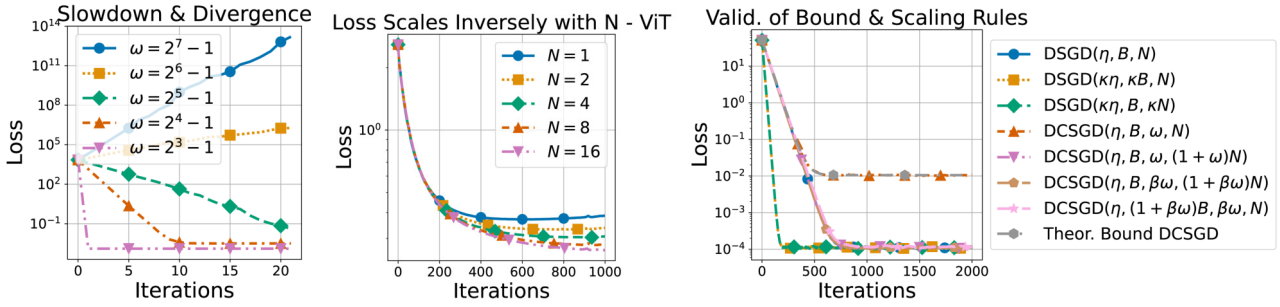

Figure 8: As per Thm. 3.7: More compression implies slowdown, up to possible divergence (Left); As per Thm. 3.7, more agents imply more optimality (Center); Validation of Bound and Scaling Rules: i) The bound derived in Thm. 3.7 matches the experimental loss of DCSGD(η, B, ω, N); ii) Consistently with Prop. 3.5, DSGD$(\kappa\eta, \kappa B, N)$ recovers the asymptotic behavior of DSGD(η, B, N); iii) DCSGD run with hyperparameters that follow Prop. 3.9 do recover the asymptotic performance of DSGD; iv) DCSGD$(\kappa\eta, \kappa B, \omega, N)$ fails to do so as it does not follow them (Right);

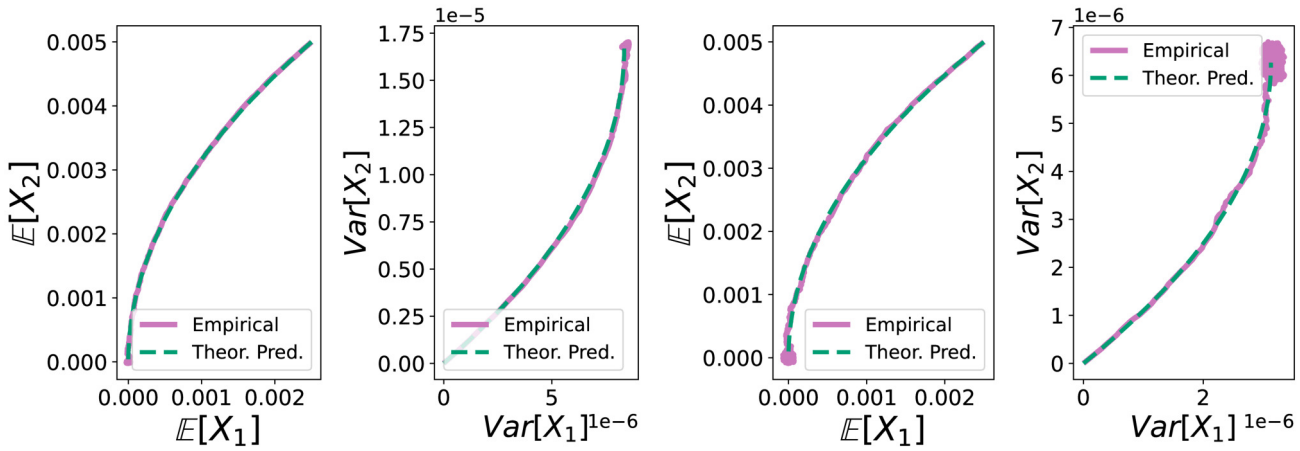

Figure 9: Verification of the Stationary Distribution of DCSGD and DSignSGD on a convex quadratic function.

Thm. 3.7 and confirm that they indeed match loss as prescribed. Additionally, we also verify the Scaling Rules as per Prop. 3.9.

F.7 DCSGD & DSignSGD - Stationary Distributions (Figure 9)

On the left of Figure 9, we validate the Stationary Distribution of DCSGD run with Rand-k while on the right we do the same for DSignSGD.

DCSGD We optimize the function $f(x) = \frac{x^\top H x}{2}$ where $H = \text{diag}(2, 1, 1, 1, 1, 1, 1, 1, 1, 1)$. We run DCSGD with Rand-k for 1000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.1$. The learning rate is $\eta = 0.01$, $k = 3$. We plot the evolution of the average variances with the theoretical predictions of Prop. C.13: Results are averaged over 50000. The experimental moments match the theoretical predictions.

DSignSGD We optimize the function $f(x) = \frac{x^\top H x}{2}$ where $H = \text{diag}(2, 1, 1, 1, 1, 1, 1, 1, 1, 1)$. We run DSignSGD for 10000 epochs as we calculate the full gradient and inject it with Gaussian noise $Z \sim \mathcal{N}(0, \sigma^2 I_d)$ where $\sigma = 0.1$. The learning rate is $\eta = 0.001$. We plot the evolution of the average variances with the theoretical predictions of Prop. D.16: Results are averaged over 5000. The experimental moments match the theoretical predictions.

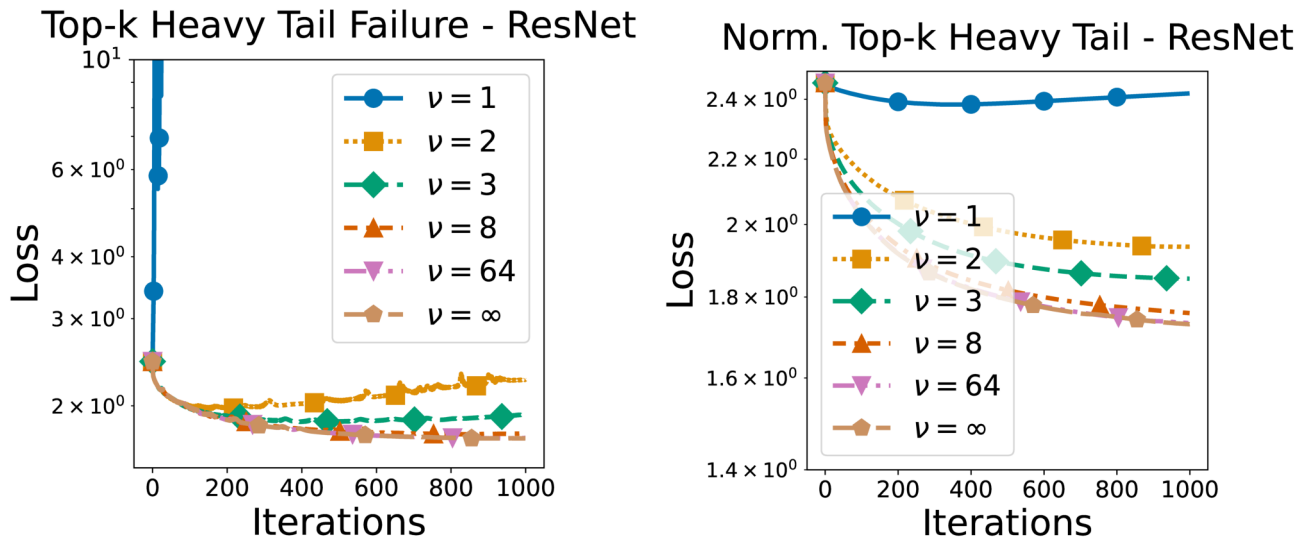

Figure 10: On the left, Top-k fails at handling increasingly heavy-tailed noise, while on the right we see that combining Top-k with Normalized SGD is promising.

F.8 Top-k and its Modification - Resilience to Heavy-Tailed Noise (Figure 10)

To produce Figure 10, we train the ResNet above on CIFAR-10. As above, we inject heavy-tailed noise onto the full gradients and observe that Top-k cannot handle such noise. However, using Top-k on top of Normalized SGD seems to mitigate this issue. Therefore, we confirm that sign compression is not the only one that can handle heavy-tailed noise and that there is room to develop alternative optimizers.

F.9 Heavy Tailed and Large Noise (Figure 11)

DCSGD - Large Noise - Top Left We optimize the function $f(x) = \frac{x^\top Hx}{2}$ where $H = I_{10}$ with DCSGD with Rand-k where $k = 1$, $\eta = 0.01$, and as we inject noise distributed as a Gaussian with covariance matrix $\Sigma = \sigma^2 I_d$ and $\sigma^2 \in \{10^{-4}, 10^{-2}, 10^0, 10^2, 10^4\}$, and $N = 3$. As the variance increases, the optimizer diverges more and more: This confirms that indeed DCSGD cannot handle large noise as its loss level scales quadratically in the noise level. The asymptotic loss level matches that predicted in Thm. 3.7. Results are averaged over 100 runs.

DSignSGD - Large Noise - Top Right We optimize the function $f(x) = \frac{x^\top Hx}{2}$ where $H = I_{10}$ with DSignSGD as we inject noise distributed as a Gaussian with covariance matrix $\sigma^2 \in \{10^{-4}, 10^{-2}, 10^0, 10^2, 10^4\}$, $\eta = 0.001$, and $N = 3$. As the variance increases, the optimizer never diverges: This confirms that indeed DSignSGD can handle large noise as its loss level scales linearly in the noise level. The asymptotic loss level matches that predicted in Thm. 3.12. Results are averaged over 100 runs.

DCSGD - Heavy-Tailed Noise - Bottom Left We optimize the function $f(x) = \frac{x^\top Hx}{2}$ where $H = I_{10}$ with DCSGD with Rand-k where $k = 1$, $\eta = 0.01$, and as we inject noise distributed as a Gaussian with covariance matrix $\Sigma = \sigma^2 I_d$ and $\sigma = 0.1$, $\nu \in \{1, 2, 3, 8, 64, \infty\}$, and $N = 3$. Even if the scale is small: 1) When $\nu = 1$, the optimizer diverges; 2) When $\nu = 2$, the loss is non-stationary; 3) The larger ν, the more stable and optimal the loss. This confirms that indeed DCSGD cannot handle Heavy-Tailed noise. Results are averaged over 100 runs.

DSignSGD - Heavy-Tailed Noise - Bottom Right We optimize the function $f(x) = \frac{x^\top Hx}{2}$ where $H = I_{10}$ with DSignSGD as we inject noise distributed as a Student's t with scale $\Sigma = \sigma^2 I_d$ and $\nu \in \{1, 2, 3, 8, 64, \infty\}$, and $N = 3$. Even if the scale is large ($\sigma = 1$) noise is of unbounded expected value, DSignSGD never diverges. Of course, fatter tails imply larger loss: This confirms that indeed DSignSGD can handle Heavy-Tailed noise. Results are averaged over 100 runs.

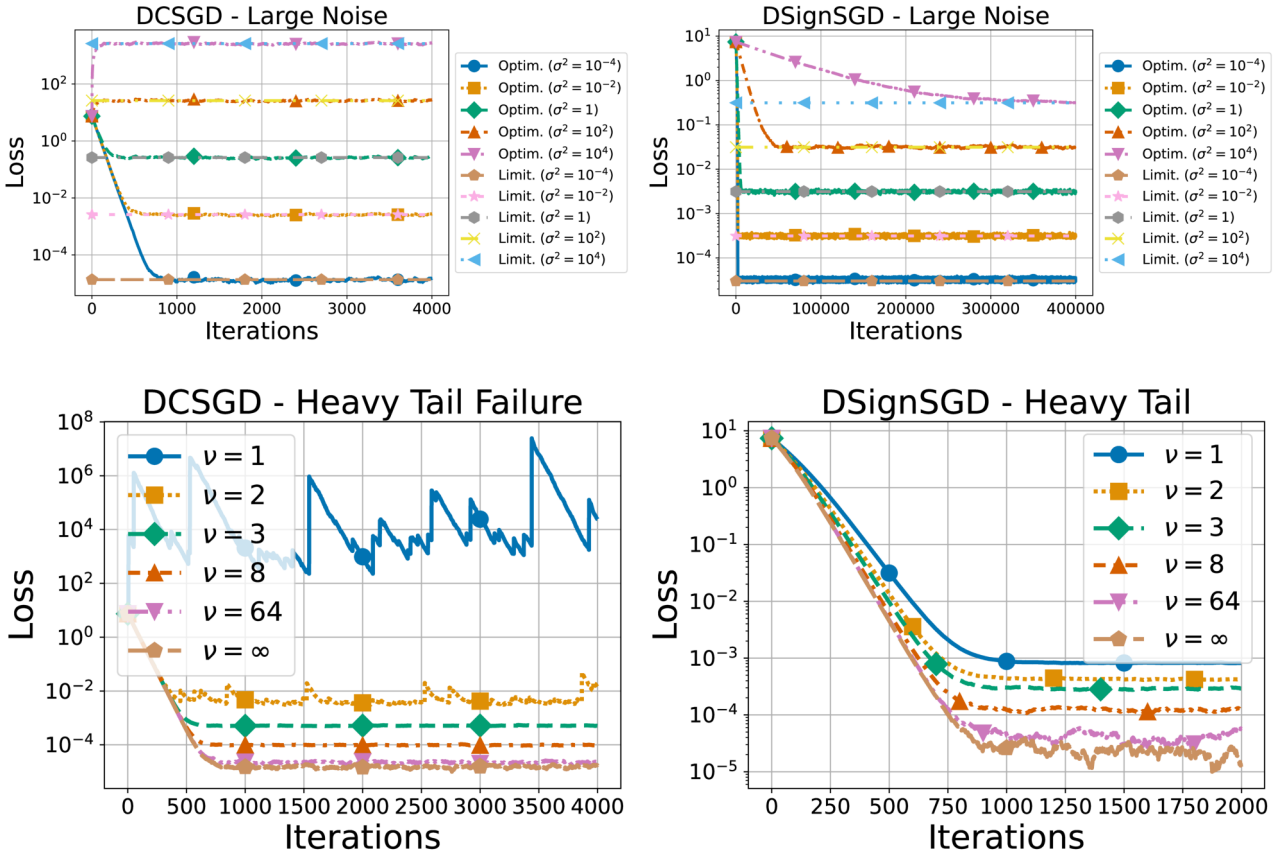

Figure 11: DCSGD cannot handle large noise as its asymptotic loss level scales quadratically in the noise level (Top Left); on the contrary, DSignSGD can as its level scales linearly in the noise level (Top Right); DCSGD cannot handle heavy-tailed noise (Bottom Left) while DSignSGD can (Bottom Right).

G Scaling Rule Validation - GPT2

This experiment aims to validate some of the scaling rules derived for DCSGD (see Table 1 related to Proposition 3.9) and DSignSGD (see Proposition 3.14) using a GPT-2-like model. To be precise, in these experiments, we fix a *base run* optimizer: Optimizer($\mathcal{H}$), where $\mathcal{H}$ is a configuration of hyperparameters (e.g., the learning rate η, the batch size B, or the number of agents N, i.e., GPUs). Then, we run the same optimizer with other hyperparameter configurations ($\tilde{\mathcal{H}}$). We verify that hyperparameter configurations ($\tilde{\mathcal{H}}$) that satisfy the functional relationships prescribed by our propositions achieve a performance much closer to the base run ($\mathcal{H}$) than those configurations that violate such prescriptions. This demonstrates that when adjustments to hyperparameters are necessary to accommodate new scenarios, one can follow our scaling rules to preserve the performance of DSignSGD/DCSGD without needing to repeat the fine-tuning process. For example, one might desire larger batch sizes to fully utilize newly available larger GPUs, or face a reduction in available GPUs due to budget cuts. We highlight that, in general, scaling rules are not meant to be exact prescriptions, but rather to give a principled approach to reduce the hyperparameter search space.

G.1 Model Architecture and Dataset

The model architecture is provided in the popular GitHub repository nanoGPT by Andrej Karpathy: All details can, of course, be found on the repository — We used the smallest configuration with 124M parameters. Regarding the dataset, we train our models on the FineWeb-Edu dataset. To do so, we minimize the *Cross-entropy* loss for $10,000$ iterations. Note that we use no learning rate schedulers, as we do not aim for optimal performance but rather for clear and fair experimental validation of our theoretical insights. We encourage future work to explore the validity of our theory in larger and more realistic settings.

G.1.1 DSignSGD - (Figure 12)

In this experiment, we fix the "**base run**" optimizer DSignSGD(η, B, N) by selecting $\eta = 0.001$, $B = 4$, and $N = 4$. We selected 6 different hyperparameter configurations: 3 that satisfy our scaling rules and 3 that do not. We ran each configuration 5 times and computed the average absolute percentage error of the last 100 iterations with respect to the "**base run**". Figure 12 shows the boxplots of these errors, and one can see that configurations that satisfy our scaling rules (marked in green in the figure) achieve a significantly lower error compared to those that do not follow them (marked in red).

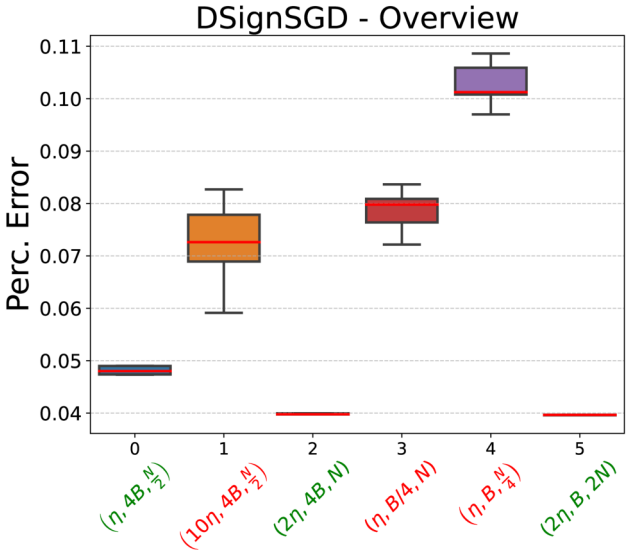

Figure 12: Boxplots of errors: Validation of Scaling Rules for DSignSGD on a 124M GPT2 model.

G.1.2 DCSGD - (Figure 13)

In this experiment, we replicate the effort above by optimizing the model with DSGD(η, B, N) as a "**base run**" (i.e., no compression). We select $\eta = 0.1$ and $B = N = 1$. Then, we run DCSGD with the Rand-k compressor; in this case, the compression factor ω is equal to $\frac{d}{k}$, where d is the total number of trainable parameters and k is the number of parameters that at each iteration are randomly selected to be trained — The remaining $d - k$ are left unchanged for that iteration. We simulated 12 different configurations and computed the average absolute percentage error of the last 100 iterations with respect to the "**base run**". Since Table 1 contains many more rules than DSignSGD, we opt for a different style of visualization that showcases the validity of multiple rules simultaneously. Since we had to simulate many more configurations, we only ran each configuration 3 times.

Left of Figure 13: In this figure, we show that: 1) For fixed (η, B, ω), increasing the number of agents N helps mitigate the performance loss due to compression (Rule 1 in green); 2) For fixed (η, N, ω), increasing the batch size B helps mitigate the performance loss due to compression (Rule 2 in orange); 3) Combining these two rules is even better (Rule 3).

Right of Figure 13: In this figure, we ran DCSGD while doubling the learning rate ($\eta = 0.2$), thereby combining the effects of compression and increased learning rate on performance. Once again, we observe that increasing compression leads to worse performance, even more so when the learning rate has doubled. However, in accordance with our scaling rules, for any compression level, increasing N and B helps mitigate the performance loss.

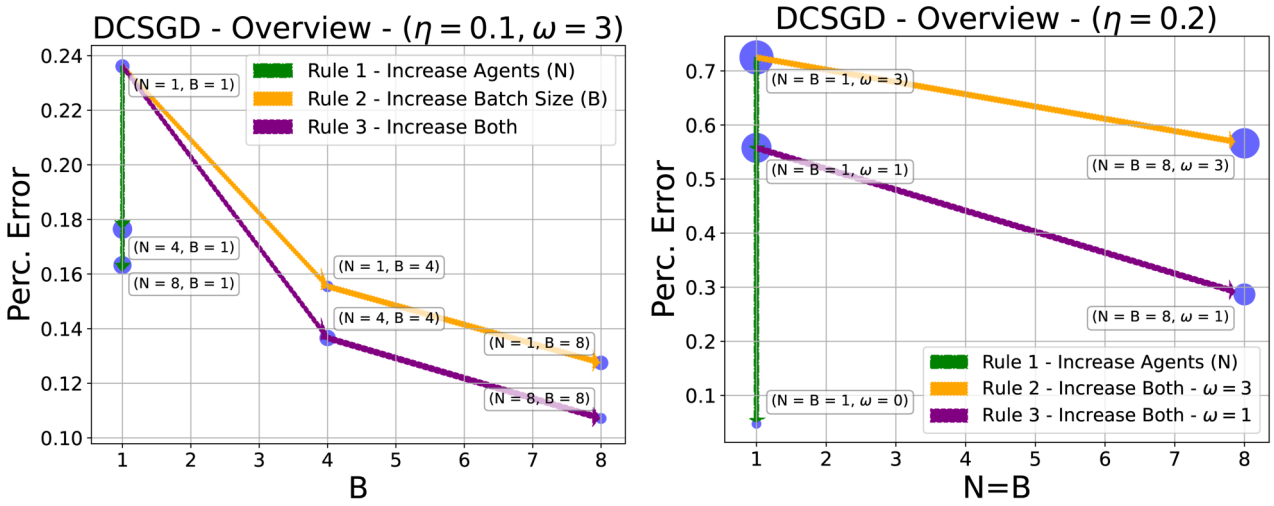

Figure 13: Validation of Scaling Rules for DCSGD on a 124M GPT2 model.