DPA: Decentralized Primal Averaging with Quasi-Global Momentum for Highly Heterogeneous Data

Diyako Ghaderyan* **Stefan Werner†**

Abstract

Decentralized training of deep learning models keeps training data local and supports on-device learning over networks. In realistic scenarios, heterogeneity across agents' local data poses an optimization challenge and can severely degrade test accuracy, especially on sparse communication graphs. Building on recent advances in centralized primal averaging, we propose Decentralized Primal Averaging (DPA), a decentralized learning algorithm that combines primal averaging with a local quasi-global momentum buffer. DPA maintains three coupled sequences for optimization, gradient evaluation, and output averaging, while using consecutive mixed iterates to construct a network-aware momentum direction without transmitting an auxiliary tracking variable. For smooth nonconvex objectives, we establish a convergence bound for DPA and identify conditions for linear speedup in the number of agents for sufficiently many iterations. We also introduce DPA-1G, a variant that uses one communication round per iteration and matches the per-iteration communication cost of DSGD-style baselines. Experiments following the GUT decentralized image-classification benchmark show that DPA compares favorably with DSGD, gradient tracking, QG-DSGDm, momentum tracking, and QG-GUTm across datasets, architectures, topologies, and heterogeneity levels, with the largest gains under high heterogeneity on sparse rings. At the same total communication cost, both DPA variants achieve higher mean test accuracy than the best baseline on CIFAR-10 and CIFAR-100 with ResNet-20. Our code is publicly available.[3]

1 Introduction

Training modern machine learning models, especially deep neural networks, often requires large datasets and substantial computational resources. This has motivated distributed training methods, where the workload is split across many agents. Early work on decentralized optimization [Tsitsiklis, 1984] and gossip-style protocols [Xiao and Boyd, 2004] showed how agents can reach agreement without a central coordinator. More recent work has shown that decentralized stochastic gradient methods can be effective for large-scale machine learning [Lian et al., 2017]. In contrast, the classical master–worker setup relies on a centralized parameter server, where workers read model parameters and send gradient updates to shared server shards [Dean et al., 2012]. Although this approach is effective, the server can become a communication bottleneck as the number of workers increases.

Decentralized learning provides an alternative that avoids a central server. It is also useful when data should remain local [Nedić, 2020]. In this setting, agents are connected through a commu-

*Department of Information and Communications Engineering, Aalto University, 02150 Espoo, Finland. E-mail: diyako.ghaderyan@aalto.fi.

†Department of Electronic Systems, Faculty of Information Technology and Electrical Engineering, Norwegian University of Science and Technology, 7034 Trondheim, Norway; also with the Department of Information and Communications Engineering, Aalto University, 02150 Espoo, Finland. E-mail: stefan.werner@ntnu.no.

[3] https://github.com/Diyako754/DPA

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

nication graph and exchange messages only with their neighbors. This reduces the per-iteration communication to local peer-to-peer exchanges [Assran et al., 2019]. Under standard assumptions, decentralized stochastic gradient methods can achieve the same $\mathcal{O}(1/\sqrt{nT})$ rate as centralized mini-batch SGD, giving a linear speedup with respect to the number of agents n for sufficiently many iterations [Koloskova et al., 2020, Lian et al., 2017].

A key step in these methods is gossip averaging. At each iteration, every agent mixes its local model with the models received from neighboring agents, using weights defined by the communication graph [Xiao and Boyd, 2004]. Since this mixing is local, information does not spread through the whole network immediately. As a result, the local models can differ from each other during training, especially in the early iterations. The communication graph therefore affects both the communication cost and the convergence behavior. Dense graphs spread information faster but require more communication per agent, while sparse graphs require less communication per agent but usually mix more slowly [Ghaderyan et al., 2024, Pu and Nedić, 2021].

The problem becomes harder when the data are heterogeneous across agents. In many practical settings, different agents observe data from different distributions. Then the local objectives may be quite different, and the local gradients may point in different directions. This can increase model drift, slow down training, and reduce test accuracy [Lin et al., 2021, Hsieh et al., 2020]. Several methods have been proposed to reduce this effect. Gradient tracking (GT) [Di Lorenzo and Scutari, 2016, Nedić et al., 2017, Koloskova et al., 2021] and related methods such as D^2 [Tang et al., 2018] correct local updates using information that better approximates the global descent direction. These methods can obtain convergence guarantees that are less affected by data heterogeneity. However, gradient-tracking methods usually require each agent to communicate an extra model-sized tracking variable, which increases the per-iteration communication cost. Other methods, such as QG-DSGDm [Lin et al., 2021] and RelaySum [Vogels et al., 2021], avoid this extra communication by changing the local update or the communication scheme. QG-DSGDm preserves DSGD's per-iteration communication cost. RelaySum has the same communication cost as gossip averaging on the same spanning tree.

Motivated by these limitations, we propose a decentralized learning algorithm that combines two ideas. First, inspired by primal averaging and schedule-free optimization [Defazio et al., 2025, 2024], we maintain separate sequences for optimization, gradient evaluation, and output averaging. This allows the output model to move more smoothly than the noisy local optimization sequence. Second, inspired by quasi-global momentum [Lin et al., 2021], we build a local quasi-global momentum buffer from consecutive mixed iterates. This buffer reflects the effect of network mixing, but does not require agents to send an extra momentum or tracking buffer. The resulting method, *Decentralized Primal Averaging* (DPA), is designed to improve training under heterogeneous data using communication between neighbors. Our main contributions are as follows.

1.1 Contributions

- We propose *Decentralized Primal Averaging* (DPA), which combines primal averaging [Defazio et al., 2024, 2025] with local quasi-global momentum [Lin et al., 2021] for decentralized learning with heterogeneous data. DPA maintains separate optimization, gradient-evaluation, and output sequences, and uses a second communication round to mix the locally averaged models while keeping the momentum buffer local.

- For smooth nonconvex objectives, we establish a convergence bound for the expected squared gradient norm at DPA's network-average output model, averaged over iterations. The bound explicitly accounts for network connectivity and data heterogeneity and identifies conditions for linear speedup in the number of agents.

- We introduce DPA-1G, a variant that uses one communication round per iteration, matching DSGD's per-iteration communication cost. It periodically replaces a stochastic-gradient update with gossip averaging of the output models.

- On CIFAR-10 and CIFAR-100 with ResNet-20 on a 16-agent ring ($\alpha = 0.01$), DPA reaches high test accuracy in fewer epochs than QG-DSGDm and QG-GUTm, using a constant learning rate after linear warmup. At the full matched communication budget, both DPA variants outperform the strongest baseline, with DPA gains of 7.48 and 16.03 percentage points in mean test accuracy, respectively (Appendix D.6).

1.2 Related work

Decentralized Stochastic Gradient Descent (DSGD) [Lian et al., 2017] is one of the standard methods for peer-to-peer distributed learning. It alternates between local stochastic-gradient steps and gossip averaging steps [Xiao and Boyd, 2004], and has been widely used for training deep models over decentralized networks [Ying et al., 2021, Lian et al., 2017]. Stochastic Gradient Push (SGP) [Assran et al., 2019] extends this idea to directed and time-varying communication graphs. A unified analysis of decentralized SGD with time-varying topologies and local updates was given in [Koloskova et al., 2020]. For nonconvex problems, NEXT [Di Lorenzo and Scutari, 2016] introduced a tracking mechanism for in-network optimization, which later became a key idea behind gradient-tracking methods for decentralized optimization [Koloskova et al., 2021, Pu and Nedić, 2021]. AdGT [Ghaderyan and Werner, 2026] combines gradient tracking with adaptive per-agent learning rates based on local smoothness and a common safety factor. It reduces tuning and often converges faster than gradient tracking with a tuned constant learning rate, without extra communication rounds. The paper also provides a linear convergence guarantee.

A common assumption in the analysis and use of DSGD is that the data are independently and identically distributed (IID) across agents. This assumption is often unrealistic in practical decentralized systems [Hsieh et al., 2020]. In many applications, each agent observes data from a different distribution, which is usually called the heterogeneous or non-IID setting. Data heterogeneity makes the local objectives differ across agents, increases the variation among local gradients, and can lead to unstable training and lower test accuracy [Lin et al., 2021]. Several methods have been proposed to reduce this effect, including D^2 [Tang et al., 2018] and gradient tracking [Nedić et al., 2017, Di Lorenzo and Scutari, 2016, Koloskova et al., 2021]. These methods correct local updates using extra information and can obtain convergence guarantees that do not depend directly on the heterogeneity parameter ζ^2 [Koloskova et al., 2021]. However, gradient-tracking methods usually require each agent to communicate both the model parameters and an additional tracking variable, which roughly doubles the communication cost per iteration compared with vanilla DSGD.

Momentum-based methods provide another approach to decentralized training with heterogeneous data. A direct momentum extension, DSGD with momentum (DSGDm) [Gao and Huang, 2020], can perform worse than DSGD when the data are heterogeneous, because local momentum buffers can increase model drift. Synchronizing the momentum buffers can improve performance [Assran et al., 2019, Yu et al., 2019], and SlowMo [Wang et al., 2020] uses a slow momentum update on globally averaged parameters. However, these approaches require extra communication. QG-DSGDm [Lin et al., 2021] avoids this by using quasi-global momentum, which estimates a global update direction from consecutive mixed iterates. RelaySum [Vogels et al., 2021] takes a different approach by changing the communication scheme itself, and can be combined with other optimization methods. Momentum tracking (MT) [Takezawa et al., 2023] combines momentum with gradient tracking and has a convergence rate independent of data heterogeneity in the nonconvex stochastic setting. QG-GUTm [Aketi et al., 2023] combines quasi-global momentum with global update tracking, which tracks model updates instead of gradients. This gives a tracking-like correction without extra communication and improves over QG-DSGDm under the same per-iteration communication cost on heterogeneous data.

2 Algorithm and setup

We consider a network of n agents connected by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. An edge $\{i, j\} \in \mathcal{E}$ means that agents i and j can directly exchange messages. We denote by $\mathcal{N}(i)$ the neighbors of agent i, including i itself. Let $W \in \mathbb{R}^{n\times n}$ be the mixing matrix associated with the graph. We assume that W is symmetric and doubly stochastic [Lian et al., 2017, Koloskova et al., 2020]. Communication is synchronous.

Each agent i has a local objective $f_i : \mathbb{R}^d \to \mathbb{R}$. The goal is to minimize the average objective

$$\min_{x\in\mathbb{R}^d} \; f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x), \qquad f_i(x) = \mathbb{E}_{\xi_i\sim\mathcal{D}_i}\big[F_i(x;\xi_i)\big], \tag{1}$$

where F_i is the loss on one data sample, ξ_i is sampled from the local distribution $\mathcal{D}_i$, and f_i is the expected local loss. At each iteration, agent i can compute a stochastic gradient $\nabla F_i(x;\xi_i)$. We assume that these gradients are unbiased and have bounded variance, as stated in Assumption 2.

Algorithm 1 Decentralized Primal Averaging (DPA)

Input: Initial model x_i^0 at agent i, $y_i^0 = x_i^0$, $u_i^0 = 0$, number of iterations T, learning rate η, momentum parameters $\beta_1, \beta_2 \in (0,1)$, schedules $\{a_k\}_{k\geq 1}$ and $\{b_k\}_{k\geq 1}$, mixing weights $\{w_{ij}\}_{j\in\mathcal{N}(i)}$, and neighbor set $\mathcal{N}(i)$ including i.

procedure AGENT-i
  for $k = 0, 1, \ldots, T-1$ **do**
    $z_i^k = b_{k+1} y_i^k + (1 - b_{k+1}) x_i^k$
    Sample $\xi_i^k \sim \mathcal{D}_i$ and compute $g_i^k = \nabla F_i(z_i^k; \xi_i^k)$
    $v_i^k = y_i^k - \eta(\beta_1 u_i^k + g_i^k)$
    SENDRECEIVE(v_i^k)
    $y_i^{k+1} = \sum_{j\in\mathcal{N}(i)} w_{ij} v_j^k$
    $d_i^{k+1} = (y_i^k - y_i^{k+1})/\eta$
    $u_i^{k+1} = (1 - \beta_2) u_i^k + \beta_2 d_i^{k+1}$
    $s_i^k = (1 - a_{k+1}) x_i^k + a_{k+1} y_i^{k+1}$
    SENDRECEIVE(s_i^k)
    $x_i^{k+1} = \sum_{j\in\mathcal{N}(i)} w_{ij} s_j^k$
  end for
  return x_i^T
end procedure

2.1 Decentralized Primal Averaging

We propose *Decentralized Primal Averaging* (DPA), a decentralized learning algorithm based on primal averaging. The use of separate sequences for optimization, gradient evaluation, and output averaging follows ideas from primal-averaging and schedule-free methods [Lan, 2012, Defazio et al., 2024]. The closest centralized method is Generalized Primal Averaging (GPA) [Defazio et al., 2025], which separates the point where gradients are evaluated from the output sequence in a centralized setting. ScheduleFree+ [Defazio, 2026], released after our submission, extends centralized schedule-free learning to large language models using inner momentum, a modified Polyak learning rate, and warm-started averaging.

DPA adapts primal averaging to decentralized non-IID learning. This introduces two challenges. First, the output models must stay close across agents. A purely local averaging step need not keep the output models close when the data are heterogeneous. We therefore use a second communication round to mix the locally averaged model

$$s_i^k = (1 - a_{k+1}) x_i^k + a_{k+1} y_i^{k+1}.$$

Second, different local objectives can still lead agents to update their models in different directions. To reduce this drift, DPA uses a local quasi-global momentum buffer u_i^k [Lin et al., 2021]. Each agent updates this buffer from the difference between its consecutive mixed iterates y_i^k and y_i^{k+1}, without sending the buffer to its neighbors.

The nonconvex analysis uses a Lyapunov argument to control the consensus errors of X_k, Y_k, and U_k, and to bound the expected squared gradient norm, averaged over iterations, at $\bar{z}_k$, the network average of the local gradient-evaluation points. These consensus errors do not arise in centralized primal averaging.

DPA updates We now describe the DPA update. The full procedure is shown in Algorithm 1. At iteration k, agent i first forms the gradient-evaluation point

$$z_i^k = b_{k+1} y_i^k + (1 - b_{k+1}) x_i^k.$$

The coefficient b_{k+1} controls where the stochastic gradient is evaluated. Larger values place z_i^k closer to the current optimization iterate y_i^k, while smaller values place it closer to the output model x_i^k. Agent i then computes

$$g_i^k = \nabla F_i(z_i^k; \xi_i^k)$$

and takes a local update step to obtain

$$v_i^k = y_i^k - \eta(\beta_1 u_i^k + g_i^k).$$

After one communication round, the optimization sequence is updated as

$$y_i^{k+1} = \sum_{j \in \mathcal{N}(i)} w_{ij} v_j^k.$$

The local quasi-global momentum buffer is then updated by

$$u_i^{k+1} = (1-\beta_2)u_i^k + \beta_2 \frac{y_i^k - y_i^{k+1}}{\eta}.$$

This buffer is not communicated. It depends on the mixed iterate y_i^{k+1}, so it reflects the effect of the network mixing without sending an extra tracking variable. Finally, agent i updates the output sequence. It first forms

$$s_i^k = (1-a_{k+1})x_i^k + a_{k+1} y_i^{k+1},$$

and then uses a second communication round to obtain x_i^{k+1}. The averaging coefficient a_{k+1} controls the weights in the local averaging update. Smaller values give greater weight to the previous output model x_i^k, while larger values give greater weight to the updated optimization iterate y_i^{k+1}.

Communication cost DPA uses two communication rounds per iteration. The first round mixes the locally updated model v_i^k, and the second round mixes the locally averaged model s_i^k. Thus, each iteration communicates two model-sized vectors to each neighbor. The momentum buffer u_i^k is kept local and is never sent to neighboring agents. In Section 2.2, we introduce DPA-1G, a variant that uses only one communication round per iteration.

Compact form Stack the local variables as $X_k = [x_1^k, \dots, x_n^k]$, and define Y_k, Z_k, U_k in the same way. Let

$$G_k = [\nabla F_1(z_1^k; \xi_1^k), \dots, \nabla F_n(z_n^k; \xi_n^k)].$$

Then DPA can be written as

$$Z_k = b_{k+1} Y_k + (1-b_{k+1}) X_k, \tag{2}$$

$$Y_{k+1} = \big(Y_k - \eta(\beta_1 U_k + G_k)\big) W, \tag{3}$$

$$U_{k+1} = (1-\beta_2)U_k + \beta_2 \frac{Y_k - Y_{k+1}}{\eta}, \tag{4}$$

$$X_{k+1} = \big((1-a_{k+1})X_k + a_{k+1} Y_{k+1}\big) W, \tag{5}$$

with $Y_0 = X_0$ and $U_0 = 0$. Write $\bar{x}_k = n^{-1} X_k \mathbf{1}$, and analogously for the other sequences.

Remark 1 (Difference from centralized primal averaging). DPA differs from centralized primal-averaging methods such as schedule-free optimization and GPA in both the optimization setting and the proof technique. Those methods maintain a single set of iterates, so they do not need to control consensus error across agents. Their theoretical guarantees are mainly based on convex stochastic optimization and online-to-batch or regret arguments, whereas our analysis treats the decentralized nonconvex setting. DPA runs on a sparse peer-to-peer graph with heterogeneous local data. The algorithm must therefore control consensus error in the optimization sequence, the gradient-evaluation sequence, the output sequence, and the local quasi-global momentum buffer. This leads to a decentralized nonconvex Lyapunov analysis that controls the consensus errors of X_k, Y_k, Z_k, and U_k, rather than the centralized online-to-batch arguments used in earlier primal-averaging work.

Remark 2 (Role of quasi-global momentum in DPA). DPA also differs from quasi-global momentum methods such as QG-DSGDm and QG-GUTm. In those methods, the quasi-global momentum buffer helps reduce drift under heterogeneous data. In DPA, primal averaging separates the local optimization iterates and the gradient-evaluation points from the output model returned by each agent. The quasi-global momentum buffer adds another correction on top of this structure. Therefore, setting $\beta_1 = 0$ removes the momentum term but does not reduce DPA to vanilla DSGD; the primal-averaging updates remain active.

Effect of momentum and the b_k schedule Appendix D.4 studies the effect of quasi-global momentum and the b_k schedule. In Figure 2, we compare DPA with several variants on CIFAR-10/ResNet-20 using a ring topology with $n = 16$ agents and $\alpha = 0.01$. Full DPA, which uses both quasi-global momentum and a decreasing b_k schedule, has higher test accuracy early in training and at the last epoch than the variants without momentum.

Among the variants without momentum, the decreasing b_k schedule performs slightly better than the fixed choices of b_k tested here. Training with $b_k = 0.2$ is unstable in this experiment. These results support using a decreasing coefficient in this setting.

2.2 Variant with one communication round per iteration

DPA uses two communication rounds per iteration, with one for the optimization sequence and one for the output sequence. To match the per-iteration communication cost of DSGD, QG-DSGDm, and QG-GUTm, we introduce DPA-1G, which performs exactly one SENDRECEIVE operation per iteration.

On most iterations, agents take a local stochastic-gradient step, apply gossip averaging to the resulting optimization iterates, update the quasi-global momentum buffer, and update the output sequence locally. Every T_x iterations, the method skips the stochastic-gradient computation, keeps the optimization sequence and momentum buffer fixed, and uses the communication round to mix the output sequence. In our experiments, we set $T_x = 5$. Thus, four out of every five iterations perform stochastic-gradient updates, and the remaining iteration is gradient-free and mixes the locally averaged models across neighbors. The full pseudocode is given in Appendix D.1. The convergence analysis covers DPA, not DPA-1G.

2.3 Assumptions

The convergence analysis uses a constant learning rate $\eta > 0$ within each run. The coefficients a_{k+1} and b_{k+1} are common to all agents and chosen before sampling. They may change with k, as long as they stay within the T-independent bounds given in Appendix A.1.

We use the following standard assumptions.

Assumption 1 (Smoothness)**.** Each f_i is L-smooth: for all $x, y \in \mathbb{R}^d$ and $i \in [n]$,

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L\|x - y\|.$$

Assumption 2 (Unbiased gradients and bounded variance)**.** For every $i \in [n]$ and $x \in \mathbb{R}^d$,

$$\mathbb{E}[\nabla F_i(x;\xi_i)] = \nabla f_i(x), \qquad \mathbb{E}\|\nabla F_i(x;\xi_i) - \nabla f_i(x)\|^2 \le \sigma^2.$$

Given the pre-sampling history $\mathcal{F}_k$, both statements hold and agents sample independently.

Assumption 3 (Bounded heterogeneity)**.** For all $x \in \mathbb{R}^d$,

$$\frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(x) - \nabla f(x)\|^2 \le \zeta^2.$$

Assumption 4 (Mixing matrix)**.** W is fixed, symmetric and doubly stochastic, with $w_{ij} = 0$ for $j \notin \mathcal{N}(i)$. For some $\rho \in (0, 1)$,

$$\max\{|\lambda_2(W)|, |\lambda_n(W)|\} \le 1 - \rho.$$

Assumption 5 (Lower bound)**.** The global objective satisfies $f(x) \ge f^\star > -\infty$ for all $x \in \mathbb{R}^d$.

Assumption 6 (Consensus initialization)**.** All agents start from the same model: $X_0 = Y_0 = x_0\mathbf{1}^\top$ and $U_0 = 0$.

3 Convergence analysis

We now give the main convergence results for DPA. Since stochastic gradients are evaluated at Z_k, we first bound the expected squared gradient norm at $\bar{z}_k$, averaged over iterations. We then transfer this bound to the network-average output model $\bar{x}_k$. The transfer uses the relation

$$\bar{z}_k - \bar{x}_k = b_{k+1}(\bar{y}_k - \bar{x}_k),$$

which follows directly from the definition of Z_k.

The proof uses two main ingredients. First, we introduce an auxiliary iterate that removes the network-average quasi-global momentum term, so that the network-average dynamics resemble a centralized stochastic-gradient update. Second, we use a weighted Lyapunov function to control the consensus errors of X_k, Y_k, and U_k. Full proofs and explicit constants are given in Appendix B.

3.1 Bound at the network average of the gradient-evaluation points

The constants $\tilde{\eta}$, χ, λ_X, and λ_Y used below are defined in Appendix A.1 and Proposition 2; C_Ψ is given in (60) and is proportional to $\eta^2(4\zeta^2 + 2\sigma^2)$.

Theorem 1 (Bound on the squared gradient norm at $\bar{z}_k$)**.** *Suppose Assumptions 1–6 hold, with the coefficient conditions of Section 2.3 and $\tilde{\eta} \le 1/(8L)$, together with the condition on the momentum parameters* (66)*, the consensus learning-rate condition* (70)*, and the learning-rate condition ensuring $K_g > 0$* (34)*. Then, for every $T \ge 1$,*

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{z}_k)\|^2 \le \frac{16(f(\bar{x}_0)-f^\star)}{\tilde{\eta}\,T} + \frac{64L^2C_\Psi}{\chi\min\{\lambda_X,\lambda_Y\}} + 16\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\tilde{\eta}L\cdot\frac{\sigma^2}{n}. \tag{6}$$

In particular, with $\eta = c/\sqrt{T}$ and $0 < c \le c_0$ as in Remark 12,

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{z}_k)\|^2 \le \frac{C_{z,1}}{\sqrt{T}} + \frac{C_{z,2}}{T}, \tag{7}$$

where $C_{z,1}, C_{z,2}$ depend only on the problem, algorithm, and network parameters; explicit forms are given in Appendix B.4.

Choosing $\eta = c\sqrt{n/T}$ gives the following bound for the network average of the gradient-evaluation points.

Corollary 1 (Linear-speedup regime)**.** *Suppose all assumptions of Theorem 1 other than the learning-rate conditions hold, with the learning rate chosen as*

$$\eta = c\sqrt{\frac{n}{T}},$$

where $c > 0$ is independent of n, T. Suppose T is large enough that

$$\frac{c^2 n}{T} \le \min\left\{\frac{(1-\beta_1)^2}{64L^2},\ \frac{a_{\min}^2\beta_2^2(1-\beta_1)^4}{192L^2},\ \eta_{\mathrm{cons}}^2\right\}. \tag{8}$$

Then

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{z}_k)\|^2 \le \frac{D_1}{\sqrt{nT}} + \frac{D_2 n}{T}, \tag{9}$$

where D_1, D_2 are given in Appendix B.4, and η_{cons}^2 is the right-hand side of (70)*.*

The bound gives linear speedup when D_1 stays bounded as n grows and T is large enough that $D_2 n/T \le D_1/\sqrt{nT}$. The corresponding statement for the network-average output sequence $\bar{x}_k$ is given in Corollary 2 in Appendix C.1.

3.2 Guarantee for the network-average output model

We now move from the intermediate bound on $\bar{z}_k$ to the network-average output model $\bar{x}_k$. Since

$$\bar{z}_k - \bar{x}_k = b_{k+1}(\bar{y}_k - \bar{x}_k),$$

it is enough to control the expected squared distance between $\bar{y}_k$ and $\bar{x}_k$, averaged over iterations.

Theorem 2 (Main convergence guarantee for the network-average output model)**.** *Suppose the assumptions of Theorem 1 hold. If $\eta = c/\sqrt{T}$ with $0 < c \le c_0$ as in Remark 12, then*

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{x}_k)\|^2 \le \frac{C_{x,1}}{\sqrt{T}} + \frac{C_{x,2}}{T} + \frac{C_{x,3}}{T^{3/2}}, \tag{10}$$

where $C_{x,1}, C_{x,2}, C_{x,3}$ depend only on the problem, algorithm, and network parameters. Explicit constants and the transfer inequality from $\bar{z}_k$ to $\bar{x}_k$ are given in Appendix C.

Remark 3. Under the assumptions of Corollary 1, with $\eta = c\sqrt{n/T}$, Appendix C.1 bounds the expected squared gradient norm at $\bar{x}_k$, averaged over iterations, by $\frac{7}{3}(D_1/\sqrt{nT}+D_2 n/T)$. Therefore, the network-average output sequence $\bar{x}_k$ inherits the same linear-speedup behavior. The residual consensus-error term in (9) is no larger than the leading term for $D_1 > 0$ when

$$T \geq \left(\frac{D_2}{D_1}\right)^2 n^3.$$

For smaller T, the consensus-error term may be the larger term. Uniform speedup from this bound requires D_1 to stay bounded as n grows. Appendix C.2 extends the iteration-averaged guarantee to local output models, not the last iterate.

4 Numerical experiments

We evaluate DPA on decentralized image classification with heterogeneous data. For fair and reproducible comparisons, we use the public GUT codebase of Aketi et al. [2023]. We follow the GUT benchmark for datasets, architectures, data partitions, and communication topologies. We report results for both DPA and DPA-1G, the variant with one communication round per iteration.

4.1 Setup

Datasets and architectures We use Fashion-MNIST with LeNet-5, CIFAR-10 with VGG-11, CIFAR-10 with ResNet-20, and CIFAR-100 with ResNet-20. We use the same preprocessing and data augmentation pipelines as Aketi et al. [2023]. Additional details are given in Appendix D.2.

Heterogeneous data partitioning We partition the training data among agents using a Dirichlet distribution with concentration parameter α. Smaller values of α tend to produce more skewed label distributions across agents and therefore stronger heterogeneity. Following Aketi et al. [2023], we use $\alpha \in \{0.1, 0.01\}$ to study moderate and strong heterogeneity.

Communication topologies The main results use a ring topology with $n = 16$ agents. Additional results for other graph topologies and network sizes are reported in Appendix D.5. All graphs use symmetric doubly stochastic mixing matrices, consistent with Assumption 4.

Baselines We compare DPA and DPA-1G with five baselines: DSGD [Lian et al., 2017], GT [Nedić et al., 2017], QG-DSGDm [Lin et al., 2021], MT [Takezawa et al., 2023], and QG-GUTm [Aketi et al., 2023]. For DSGD, QG-DSGDm, and QG-GUTm, we adopt the hyperparameters from the public GUT codebase [Aketi et al., 2023], which we also tested in our experimental settings. For DPA and DPA-1G, we tune the optimizer hyperparameters while retaining the same data partitions, topology, and evaluation protocol as the baselines. The learning rates and method-specific hyperparameters are listed in Appendix D.3.

Baseline learning-rate schedule For DSGD, GT, QG-DSGDm, MT, and QG-GUTm, we follow the stepwise learning-rate decay used in the GUT benchmark [Aketi et al., 2023]: the learning rate is divided by 10 after 50% and 75% of each run's training horizon. DPA and DPA-1G do not use this decay schedule. Warmup did not improve baseline test accuracy in our tests, so we retain the baseline schedule without warmup.

Linear warmup for DPA and DPA-1G The theory uses a constant learning rate η. In DPA and DPA-1G experiments, we increase the learning rate linearly to η over T_w iterations and then keep the learning rate constant:

$$\eta_k = \begin{cases} \eta\dfrac{k+1}{T_w}, & k < T_w, \\ \eta, & k \geq T_w. \end{cases}$$

When warmup is specified as a fraction of a reference training horizon, we multiply the warmup duration in epochs by the number of mini-batches per epoch to obtain T_w. In the equal-epoch non-IID runs, we use a warmup over 25% of training for $\alpha = 0.1$ and over 50% for $\alpha = 0.01$.

Table 1: Test accuracy (%) on a 16-agent ring.

Method	Fashion-MNIST / LeNet-5	CIFAR-10 / VGG-11	CIFAR-10 / ResNet-20	CIFAR-100 / ResNet-20
	$\alpha = 0.1$			
	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std
DSGD	85.605 ± 2.962 / 85.974 ± 2.374	65.99 ± 2.07 / 65.97 ± 1.95	57.71 ± 3.08 / 57.49 ± 3.24	25.53 ± 0.48 / 26.19 ± 0.60
GT	84.517 ± 0.156 / 84.515 ± 0.154	76.13 ± 1.16 / 76.12 ± 1.17	69.90 ± 2.73 / 70.25 ± 2.33	41.95 ± 1.35 / 42.15 ± 1.47
QG-DSGDm	89.717 ± 0.230 / 89.678 ± 0.273	75.57 ± 1.85 / 75.43 ± 1.24	81.73 ± 1.58 / 81.79 ± 1.80	44.89 ± 1.85 / 44.64 ± 1.39
MT	90.064 ± 0.034 / 90.048 ± 0.050	81.90 ± 1.81 / 82.44 ± 1.18	79.83 ± 1.64 / 79.79 ± 0.96	49.18 ± 3.43 / 49.54 ± 3.87
QG-GUTm	89.667 ± 0.002 / 89.671 ± 0.042	81.32 ± 3.02 / 80.89 ± 3.71	85.34 ± 1.42 / 85.17 ± 1.30	45.04 ± 0.73 / 45.34 ± 1.11
DPA-1G	89.33 ± 0.65 / 89.30 ± 0.60	**83.62 ± 1.43** / **83.60 ± 1.46**	86.58 ± 0.85 / 86.08 ± 1.46	53.92 ± 0.55 / 53.92 ± 0.56
DPA	**90.08 ± 0.28** / **90.13 ± 0.26**	83.42 ± 1.56 / 83.46 ± 1.54	**86.73 ± 1.10** / **86.65 ± 1.21**	**54.12 ± 0.97** / **54.08 ± 0.92**
	$\alpha = 0.01$			
	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std	LE $\pm$ std / L5E $\pm$ std
DSGD	73.930 ± 2.985 / 74.310 ± 3.002	41.00 ± 3.44 / 40.72 ± 3.72	29.79 ± 4.60 / 30.26 ± 4.43	16.14 ± 0.82 / 16.14 ± 0.48
GT	84.102 ± 0.226 / 84.098 ± 0.227	70.64 ± 1.77 / 70.64 ± 1.77	58.38 ± 0.81 / 58.29 ± 0.95	34.20 ± 1.85 / 34.40 ± 1.88
QG-DSGDm	76.456 ± 3.624 / 76.977 ± 3.128	62.92 ± 2.47 / 63.12 ± 1.18	61.47 ± 1.93 / 60.27 ± 1.44	33.57 ± 0.80 / 33.14 ± 1.80
MT	82.090 ± 4.793 / 83.381 ± 3.476	65.97 ± 3.92 / 65.26 ± 3.59	48.20 ± 2.49 / 48.22 ± 0.50	30.09 ± 2.63 / 31.09 ± 3.22
QG-GUTm	76.360 ± 5.199 / 76.513 ± 5.839	75.49 ± 1.49 / 75.71 ± 1.40	66.47 ± 1.74 / 65.88 ± 2.57	35.54 ± 1.01 / 35.77 ± 1.33
DPA-1G	**85.60 ± 2.46** / **85.53 ± 2.48**	75.73 ± 2.73 / 75.67 ± 2.86	78.57 ± 4.89 / 78.50 ± 4.94	48.77 ± 1.16 / 48.75 ± 1.12
DPA	84.73 ± 3.97 / 83.15 ± 5.80	**77.06 ± 2.49** / **76.84 ± 2.54**	**82.01 ± 1.47** / **81.95 ± 1.51**	**51.97 ± 1.18** / **51.86 ± 1.17**

For DPA and DPA-1G, we use the exponential coefficient schedules described in Appendix A.2. The coefficient schedule parameters are listed in Appendix D.3.

Evaluation protocol In the equal-epoch comparisons, all methods are trained on identical hardware for 100 epochs on Fashion-MNIST and 200 on CIFAR-10 and CIFAR-100. Each agent evaluates its local model on the full test set. For each seed, we first average test accuracy across agents at the end of each epoch. Unless stated otherwise, tables report both last-epoch accuracy (LE) and the average accuracy over the last five epochs (L5E), computed within each seed. We then report the mean and standard deviation (std) across three random seeds (using a divisor of three for the variance). We use L5E as the main metric because it is less sensitive to fluctuations at the end of training than LE. Boldface indicates the highest mean accuracy for each setting, with LE and L5E compared separately. Test accuracy curves show the mean test accuracy at each epoch, with shaded regions indicating ± 1 standard deviation across seeds.

Remark 4. After training, the GUT code evaluates the models once more after $K_{\text{post}} = \min(51, \text{len}(\texttt{train_loader}))$ communication rounds of gossip averaging with no gradient updates. We refer to this phase as post-training averaging. We exclude this evaluation from LE and L5E and do not count these rounds in the communication budget. For DPA and DPA-1G, the final evaluation simply repeats the last measurement and is also excluded. Appendix D.9 reports the effect of post-training averaging separately.

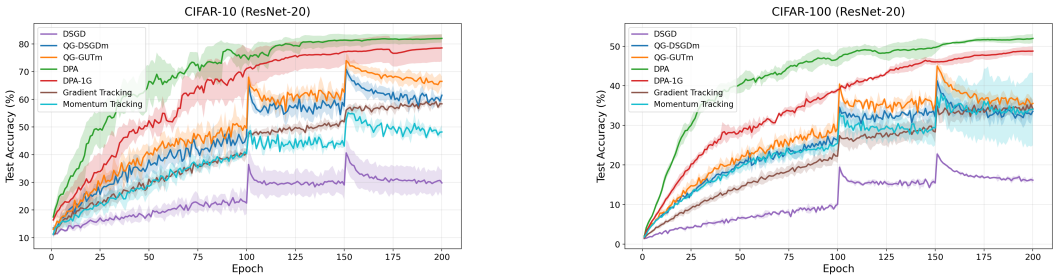
4.2 Accuracy under heterogeneous data

Table 1 reports results on the ring topology with $n = 16$ agents. Under moderate heterogeneity ($\alpha = 0.1$), DPA has the highest mean L5E accuracy on CIFAR-10/ResNet-20 (86.65%) and CIFAR-100/ResNet-20 (54.08%), while DPA-1G has the highest mean L5E accuracy on CIFAR-10/VGG-11 (83.60%). On Fashion-MNIST/LeNet-5, DPA and MT have similar mean L5E accuracies (90.13% and 90.05%, respectively).

Under stronger heterogeneity ($\alpha = 0.01$), DPA's largest gains are on the two ResNet-20 tasks. DPA has the highest mean LE and L5E on CIFAR-10/ResNet-20 and CIFAR-100/ResNet-20, improving over QG-GUTm by 16.07 and 16.09 percentage points in L5E, respectively. On CIFAR-10/VGG-11, DPA also has the highest mean LE and L5E, but its L5E gain over QG-GUTm is smaller (1.13 percentage points). With $T_x = 5$, DPA-1G has a mean L5E accuracy of 75.67%. On Fashion-MNIST/LeNet-5, DPA-1G has the highest mean LE and L5E, improving over GT, the strongest baseline, by 1.43 percentage points in L5E.

Additional results for $n = 32$ agents and other graph topologies are given in Appendix D.5. Figure 1 shows the $n = 16$ ring test accuracy curves for $\alpha = 0.01$. Among the methods that use one communication round per iteration, DPA-1G has the highest mean LE accuracy on CIFAR-100/ResNet-20.

Remark 5. DPA uses two communication rounds per iteration, so DPA sends more data per iteration than DSGD-style methods that use one communication round per iteration. However, Figure 1 shows



(a) CIFAR-10 / ResNet-20

(b) CIFAR-100 / ResNet-20

Figure 1: Test accuracy during training on a 16-agent ring ($\alpha = 0.01$).

that on the heterogeneous ring with $\alpha = 0.01$, DPA reaches high test accuracy in fewer epochs than QG-DSGDm and QG-GUTm. At the full matched communication budget, DPA's mean test accuracy exceeds that of the strongest baseline by 7.48 percentage points on CIFAR-10 and 16.03 percentage points on CIFAR-100 (Appendix D.6).

See Appendices D.7 and D.8 for IID and timing results.

5 Conclusion

We introduced *Decentralized Primal Averaging* (DPA), a decentralized learning algorithm for heterogeneous data. DPA adapts primal averaging to the gossip-based setting by using three coupled sequences for optimization, gradient evaluation, and output averaging. It also uses a local quasi-global momentum buffer built from consecutive mixed iterates, without sending an extra tracking variable. For smooth nonconvex objectives, we bounded the expected squared gradient norm at the network-average output model, averaged over iterations. The bound gives linear speedup in the number of agents when the network- and heterogeneity-dependent term is no larger than the leading term. We also introduced DPA-1G, a variant with one communication round per iteration that matches the per-iteration communication cost of DSGD-style baselines. Experiments on decentralized image classification show that DPA and DPA-1G improve over strong baselines, with the largest gains in highly heterogeneous settings on sparse ring topologies. Future work includes extending the analysis to DPA-1G, relaxing the conservative condition on the momentum parameters, and adapting primal averaging to time-varying or asynchronous networks.

Acknowledgments and Disclosure of Funding

This work was partially supported by the Research Council of Finland (Grant 354523). We acknowledge the computational resources provided by the Aalto Science-IT project.

References

Sai Aparna Aketi, Abolfazl Hashemi, and Kaushik Roy. Global update tracking: a decentralized learning algorithm for heterogeneous data. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 48939–48961, 2023.

Mahmoud Assran, Nicolas Loizou, Nicolas Ballas, and Mike Rabbat. Stochastic gradient push for distributed deep learning. In *International Conference on Machine Learning*, pages 344–353. PMLR, 2019.

Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, Marc'Aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. *Advances in Neural Information Processing Systems*, 25, 2012.

Aaron Defazio. ScheduleFree+: Scaling learning-rate-free & schedule-free learning to large language models. *arXiv preprint arXiv:2605.19095*, 2026.

Aaron Defazio, Xingyu Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled. *Advances in Neural Information Processing Systems*, 37: 9974–10007, 2024.

Aaron Defazio, Konstantin Mishchenko, Parameswaran Raman, Hao-Jun Michael Shi, and Lin Xiao. Smoothing DiLoCo with primal averaging for faster training of LLMs. *arXiv preprint arXiv:2512.17131*, 2025.

Paolo Di Lorenzo and Gesualdo Scutari. NEXT: In-network nonconvex optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2(2):120–136, 2016.

Hongchang Gao and Heng Huang. Periodic stochastic gradient descent with momentum for decentralized training. *arXiv preprint arXiv:2008.10435*, 2020.

Diyako Ghaderyan and Stefan Werner. AdGT: Decentralized gradient tracking with adaptive per-agent stepsizes. *IEEE Transactions on Signal Processing*, 74:4097–4113, 2026. doi: 10.1109/TSP.2026.3733814.

Diyako Ghaderyan, Necdet Serhat Aybat, A. Pedro Aguiar, and Fernando Lobo Pereira. A fast row-stochastic decentralized method for distributed optimization over directed graphs. *IEEE Transactions on Automatic Control*, 69(1):275–289, 2024.

Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip Gibbons. The Non-IID data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pages 4387–4398. PMLR, 2020.

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*, pages 5381–5393. PMLR, 2020.

Anastasiia Koloskova, Tao Lin, and Sebastian U Stich. An improved analysis of gradient tracking for decentralized machine learning. *Advances in Neural Information Processing Systems*, 34: 11422–11435, 2021.

Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1):365–397, 2012.

Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? A case study for decentralized parallel stochastic gradient descent. *Advances in Neural Information Processing Systems*, 30, 2017.

Tao Lin, Sai Praneeth Karimireddy, Sebastian U. Stich, and Martin Jaggi. Quasi-global momentum: Accelerating decentralized deep learning on heterogeneous data. In *ICML*, pages 6654–6665, 2021. URL `http://proceedings.mlr.press/v139/lin21c.html`.

Angelia Nedić. Distributed gradient methods for convex machine learning problems in networks: Distributed optimization. *IEEE Signal Processing Magazine*, 37(3):92–101, 2020.

Angelia Nedić, Alex Olshevsky, and Wei Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, 187(1):409–457, 2021.

Yuki Takezawa, Han Bao, Kenta Niwa, Ryoma Sato, and Makoto Yamada. Momentum tracking: Momentum acceleration for decentralized deep learning on heterogeneous data. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=8koy8QuTZD`.

Hanlin Tang, Xiangru Lian, Ming Yan, Ce Zhang, and Ji Liu. D^2: Decentralized training over decentralized data. In *International Conference on Machine Learning*, pages 4848–4856. PMLR, 2018.

John N Tsitsiklis. Problems in decentralized decision making and computation. Technical report, Massachusetts Institute of Technology, Laboratory for Information and Decision Systems, 1984.

Thijs Vogels, Lie He, Anastasiia Koloskova, Sai Praneeth Karimireddy, Tao Lin, Sebastian U Stich, and Martin Jaggi. RelaySum for decentralized deep learning on heterogeneous data. *Advances in Neural Information Processing Systems*, 34:28004–28015, 2021.

Jianyu Wang, Vinayak Tantia, Nicolas Ballas, and Michael Rabbat. SlowMo: Improving communication-efficient distributed SGD with slow momentum. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=SkxJ8REYPH`.

Lin Xiao and Stephen Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Bicheng Ying, Kun Yuan, Yiming Chen, Hanbin Hu, Pan Pan, and Wotao Yin. Exponential graph is provably efficient for decentralized deep training. *Advances in Neural Information Processing Systems*, 34:13975–13987, 2021.

Hao Yu, Rong Jin, and Sen Yang. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. In *International Conference on Machine Learning*, pages 7184–7193. PMLR, 2019.

A Algorithm details and additional notation

The problem statement, algorithm, and standing assumptions are in Section 2. This appendix gives the notation, coefficient bounds, schedule definitions, and technical conditions used in the proofs.

A.1 Coefficient bounds and notation

Throughout the analysis, we use a constant learning rate $\eta > 0$. The coefficient schedules a_{k+1} and b_{k+1} may be time-varying, provided they satisfy

$$0 < b_{\min} \le b_{k+1} \le b_{\max} < 1, \qquad 0 < a_{\min} \le a_{k+1} \le a_{\max} < 1.$$

The momentum parameters satisfy $\beta_1, \beta_2 \in (0, 1)$, and we require

$$\tilde{\eta} \coloneqq \frac{\eta}{1-\beta_1} \le \frac{1}{8L}, \tag{11}$$

together with the additional learning-rate conditions stated below when needed.

Let

$$J \coloneqq \frac{1}{n}\mathbf{1}\mathbf{1}^\top.$$

For any $A \in \mathbb{R}^{d\times n}$, define

$$\bar{a} \coloneqq \frac{1}{n}A\mathbf{1} \in \mathbb{R}^d, \qquad \bar{A} \coloneqq \bar{a}\mathbf{1}^\top = AJ.$$

Thus $\bar{X}_k = \bar{x}_k\mathbf{1}^\top$, and similarly for Y_k, Z_k, U_k, G_k. We write $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot \mid \mathcal{F}_k]$.

Remark 6. Under Assumption 4, the mixing matrix satisfies $\|W\|_2 \le 1$ and $\|I - W\|_2 \le 2$.

Remark 7. For every $A \in \mathbb{R}^{d\times n}$,

$$\left\|AW - \bar{A}\right\|_F \le (1-\rho)\left\|A - \bar{A}\right\|_F.$$

A.2 Schedules for a_{k+1} and b_{k+1}

For each coefficient $c \in \{a, b\}$, we use either a linear or an exponential schedule:

$$c_{k+1} = c_{\text{start}} + (c_{\text{end}} - c_{\text{start}}) \min\left(1, \frac{k+1}{T_c}\right) \qquad \text{(linear)},$$

or

$$c_{k+1} = c_{\text{start}} \left(\frac{c_{\text{end}}}{c_{\text{start}}}\right)^{\left(\min\left(1, \frac{k+1}{T_c}\right)\right)^{q_c}} \qquad \text{(exponential)}.$$

Here $T_c > 0$ is the schedule length and $q_c > 0$ controls the curvature of the exponential schedule. The start and end values are chosen so that the bounds in Appendix A.1 hold at every iteration.

B Proof of Theorem 1

Throughout this proof, we use the assumptions of Section 2.3 and the coefficient bounds in Appendix A.1.

B.1 Step 1: auxiliary iterate and descent

Since W is doubly stochastic, gossip averaging preserves the network average. The auxiliary iterate $\hat{x}_k$ is a shifted version of $\bar{y}_k$, chosen so that its update is driven only by the network-average stochastic gradient $\bar{g}_k$.

Lemma 1. *Let*

$$\alpha_u := 1 - \beta_2(1-\beta_1).$$

Then, for every $k \ge 0$,

$$\bar{y}_{k+1} = \bar{y}_k - \eta(\beta_1 \bar{u}_k + \bar{g}_k), \tag{12}$$
$$\bar{u}_{k+1} = \alpha_u \bar{u}_k + \beta_2 \bar{g}_k, \tag{13}$$
$$\bar{x}_{k+1} = (1 - a_{k+1})\bar{x}_k + a_{k+1}\bar{y}_{k+1}. \tag{14}$$

Proof. Since $W\mathbf{1} = \mathbf{1}$, we have $AW\frac{1}{n}\mathbf{1} = A\frac{1}{n}\mathbf{1}$ for any $A \in \mathbb{R}^{d\times n}$. Averaging the Y-update gives

$$\bar{y}_{k+1} = \bar{y}_k - \eta(\beta_1 \bar{u}_k + \bar{g}_k),$$

which proves (12). Similarly, averaging the X-update gives

$$\bar{x}_{k+1} = (1 - a_{k+1})\bar{x}_k + a_{k+1}\bar{y}_{k+1},$$

which proves (14). Moreover, writing $\bar{d}_{k+1} = n^{-1}\sum_{i=1}^n d_i^{k+1}$, (12) gives

$$\bar{d}_{k+1} = \frac{\bar{y}_k - \bar{y}_{k+1}}{\eta} = \beta_1 \bar{u}_k + \bar{g}_k.$$

Averaging the U-update then yields

$$\bar{u}_{k+1} = (1-\beta_2)\bar{u}_k + \beta_2 \bar{d}_{k+1} = \big(1 - \beta_2(1-\beta_1)\big)\bar{u}_k + \beta_2 \bar{g}_k = \alpha_u \bar{u}_k + \beta_2 \bar{g}_k.$$

This proves (13) and completes the proof. □

B.1.1 Second-moment bounds

Lemma 2. *For every $k \geq 0$,*

$$\mathbb{E}_k \left\|\bar{g}_k\right\|^2 \leq 2 \left\|\nabla f(\bar{z}_k)\right\|^2 + 2L^2 \left(\frac{1}{n} \left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{\sigma^2}{n}, \tag{15}$$

$$\mathbb{E}_k \left\|\beta_1 \bar{u}_k + \bar{g}_k\right\|^2 \leq 2\beta_1^2 \left\|\bar{u}_k\right\|^2 + 4 \left\|\nabla f(\bar{z}_k)\right\|^2 + 4L^2 \left(\frac{1}{n} \left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{2\sigma^2}{n}. \tag{16}$$

Proof. Since

$$\mathbb{E}_k[\bar{g}_k] = \frac{1}{n} \sum_{i=1}^n \nabla f_i(z_i^k),$$

the bias–variance decomposition gives

$$\begin{aligned}
\mathbb{E}_k \left\|\bar{g}_k\right\|^2 &= \mathbb{E}_k \left\|\bar{g}_k - \frac{1}{n} \sum_{i=1}^n \nabla f_i(z_i^k)\right\|^2 + \left\|\frac{1}{n} \sum_{i=1}^n \nabla f_i(z_i^k)\right\|^2 \\
&\leq \frac{\sigma^2}{n} + \left\|\frac{1}{n} \sum_{i=1}^n \nabla f_i(z_i^k)\right\|^2 \\
&\leq \frac{\sigma^2}{n} + 2 \left\|\nabla f(\bar{z}_k)\right\|^2 + 2 \left\|\frac{1}{n} \sum_{i=1}^n \left(\nabla f_i(z_i^k) - \nabla f_i(\bar{z}_k)\right)\right\|^2 \\
&\leq \frac{\sigma^2}{n} + 2 \left\|\nabla f(\bar{z}_k)\right\|^2 + 2L^2 \left(\frac{1}{n} \left\|Z_k - \bar{Z}_k\right\|_F^2\right).
\end{aligned}$$

This proves (15). The first inequality uses Assumption 2 and conditional independence across agents given $\mathcal{F}_k$; the second uses $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$; the last uses Jensen's inequality and Assumption 1.

For (16), applying $\left\|a+b\right\|^2 \leq 2 \left\|a\right\|^2 + 2 \left\|b\right\|^2$ and (15) gives

$$\begin{aligned}
\mathbb{E}_k \left\|\beta_1 \bar{u}_k + \bar{g}_k\right\|^2 &\leq 2\beta_1^2 \left\|\bar{u}_k\right\|^2 + 2\mathbb{E}_k \left\|\bar{g}_k\right\|^2 \\
&\leq 2\beta_1^2 \left\|\bar{u}_k\right\|^2 + 4 \left\|\nabla f(\bar{z}_k)\right\|^2 + 4L^2 \left(\frac{1}{n} \left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{2\sigma^2}{n}.
\end{aligned}$$

This proves (16). □

B.1.2 Definition of the auxiliary iterate

We now define the auxiliary iterate and verify the cancellation of the network-average momentum term.

Lemma 3. *Let*

$$\tilde{\eta} \coloneqq \frac{\eta}{1-\beta_1}, \qquad c_m \coloneqq \frac{\eta\beta_1}{\beta_2(1-\beta_1)}, \qquad \hat{x}_k \coloneqq \bar{y}_k - c_m \bar{u}_k.$$

Then $\hat{x}_0 = \bar{x}_0$, and for every $k \geq 0$,

$$\hat{x}_{k+1} = \hat{x}_k - \tilde{\eta}\, \bar{g}_k. \tag{17}$$

Proof. By Lemma 1,

$$\begin{aligned}
\hat{x}_{k+1} &= \bar{y}_{k+1} - c_m \bar{u}_{k+1} \\
&= \bar{y}_k - \eta(\beta_1 \bar{u}_k + \bar{g}_k) - c_m(\alpha_u \bar{u}_k + \beta_2 \bar{g}_k) \\
&= \hat{x}_k + \big(c_m(1-\alpha_u) - \eta\beta_1\big)\bar{u}_k - (\eta + c_m\beta_2)\bar{g}_k.
\end{aligned}$$

Since $1 - \alpha_u = \beta_2(1-\beta_1)$, the definition of c_m gives

$$c_m(1-\alpha_u) = \eta\beta_1, \qquad \eta + c_m\beta_2 = \tilde{\eta}.$$

Thus (17) follows. Finally,

$$\hat{x}_0 = \bar{y}_0 - c_m \bar{u}_0 = \bar{x}_0,$$

because $Y_0 = X_0$ and $U_0 = 0$. □

The update in (17) has the form of a centralized stochastic-gradient update with learning rate $\tilde{\eta}$. Using the smoothness of f, we obtain the following one-step bound. The bound accounts for the gap between $\hat{x}_k$ and $\bar{z}_k$, the consensus error of the local gradient-evaluation points, and gradient noise.

Lemma 4. *Under* (11)*, for every* $k \geq 0$*,*

$$\begin{aligned}\mathbb{E}_k f(\hat{x}_{k+1}) \leq f(\hat{x}_k) - \frac{\tilde{\eta}}{8}\left\|\nabla f(\bar{z}_k)\right\|^2 + 3\tilde{\eta}L^2(1-b_{\min})^2\left\|\bar{y}_k - \bar{x}_k\right\|^2 \\ + 3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\left\|\bar{u}_k\right\|^2 \\ + \left(\tilde{\eta}L^2 + L^3\tilde{\eta}^2\right)\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{L\tilde{\eta}^2}{2}\cdot\frac{\sigma^2}{n}. \end{aligned} \tag{18}$$

Proof. By Lemma 3, $\hat{x}_{k+1} = \hat{x}_k - \tilde{\eta}\,\bar{g}_k$. Therefore,

$$\begin{aligned}\mathbb{E}_k f(\hat{x}_{k+1}) &\overset{(a)}{\leq} f(\hat{x}_k) - \tilde{\eta}\left\langle \nabla f(\hat{x}_k), \frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)\right\rangle + \frac{L\tilde{\eta}^2}{2}\mathbb{E}_k\left\|\bar{g}_k\right\|^2 \\ &\overset{(b)}{\leq} f(\hat{x}_k) - \frac{\tilde{\eta}}{2}\left\|\nabla f(\hat{x}_k)\right\|^2 + \frac{\tilde{\eta}}{2}\left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k) - \nabla f(\hat{x}_k)\right\|^2 + \frac{L\tilde{\eta}^2}{2}\mathbb{E}_k\left\|\bar{g}_k\right\|^2 \\ &\overset{(c)}{\leq} f(\hat{x}_k) - \frac{\tilde{\eta}}{4}\left\|\nabla f(\bar{z}_k)\right\|^2 + \frac{\tilde{\eta}}{2}\left\|\nabla f(\hat{x}_k) - \nabla f(\bar{z}_k)\right\|^2 \\ &\quad + \frac{\tilde{\eta}}{2}\left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k) - \nabla f(\hat{x}_k)\right\|^2 + \frac{L\tilde{\eta}^2}{2}\mathbb{E}_k\left\|\bar{g}_k\right\|^2 \\ &\overset{(d)}{\leq} f(\hat{x}_k) - \frac{\tilde{\eta}}{4}\left\|\nabla f(\bar{z}_k)\right\|^2 + \frac{3\tilde{\eta}}{2}\left\|\nabla f(\hat{x}_k) - \nabla f(\bar{z}_k)\right\|^2 \\ &\quad + \tilde{\eta}\left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k) - \nabla f(\bar{z}_k)\right\|^2 + \frac{L\tilde{\eta}^2}{2}\mathbb{E}_k\left\|\bar{g}_k\right\|^2 \\ &\overset{(e)}{\leq} f(\hat{x}_k) - \frac{\tilde{\eta}}{4}\left\|\nabla f(\bar{z}_k)\right\|^2 + \frac{3\tilde{\eta}L^2}{2}\left\|\hat{x}_k - \bar{z}_k\right\|^2 \\ &\quad + \tilde{\eta}L^2\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{L\tilde{\eta}^2}{2}\mathbb{E}_k\left\|\bar{g}_k\right\|^2, \end{aligned} \tag{19}$$

where (a) uses Assumption 1 and $\mathbb{E}_k[\bar{g}_k] = \frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)$; (b) uses $-\langle a,b\rangle \leq -\frac{1}{2}\|a\|^2 + \frac{1}{2}\|a-b\|^2$; (c) uses $-\frac{1}{2}\|a\|^2 \leq -\frac{1}{4}\|b\|^2 + \frac{1}{2}\|a-b\|^2$; (d) uses $\|p-r\|^2 \leq 2\|p-q\|^2 + 2\|q-r\|^2$; and (e) uses Assumption 1 together with the bound derived using Jensen's inequality in the proof of Lemma 2.

By definition of $\hat{x}_k$ and $\bar{z}_k$,

$$\hat{x}_k - \bar{z}_k = (1-b_{k+1})(\bar{y}_k - \bar{x}_k) - \frac{\eta\beta_1}{\beta_2(1-\beta_1)}\bar{u}_k,$$

so $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ and $b_{k+1} \geq b_{\min}$ give

$$\left\|\hat{x}_k - \bar{z}_k\right\|^2 \leq 2(1-b_{\min})^2\left\|\bar{y}_k - \bar{x}_k\right\|^2 + 2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\left\|\bar{u}_k\right\|^2. \tag{20}$$

Substituting (20) into (19) and applying (15) to the $\mathbb{E}_k\left\|\bar{g}_k\right\|^2$ term yields

$$\begin{aligned}\mathbb{E}_k f(\hat{x}_{k+1}) \leq f(\hat{x}_k) - \left(\frac{\tilde{\eta}}{4} - L\tilde{\eta}^2\right)\left\|\nabla f(\bar{z}_k)\right\|^2 \\ + 3\tilde{\eta}L^2(1-b_{\min})^2\left\|\bar{y}_k - \bar{x}_k\right\|^2 + 3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\left\|\bar{u}_k\right\|^2 \\ + \left(\tilde{\eta}L^2 + L^3\tilde{\eta}^2\right)\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{L\tilde{\eta}^2}{2}\cdot\frac{\sigma^2}{n}. \end{aligned}$$

Since $\tilde{\eta} \le 1/(8L)$, $\tilde{\eta}/4 - L\tilde{\eta}^2 \ge \tilde{\eta}/8$, proving (18). □

The one-step bound in Lemma 4 contains $\|\bar{y}_k - \bar{x}_k\|^2$ and $\|\bar{u}_k\|^2$. The next two lemmas give recursions for these quantities. Proposition 1 then bounds the expected squared gradient norm, averaged over iterations, in terms of the consensus error of Z_k, the initial objective gap, and the gradient variance.

Lemma 5. *For every $k \ge 0$,*

$$\mathbb{E}_k\|\bar{y}_{k+1} - \bar{x}_{k+1}\|^2 \le (1 - a_{\min})\|\bar{y}_k - \bar{x}_k\|^2 + \frac{(1-a_{\min})^2}{a_{\min}}\eta^2\left(2\beta_1^2\|\bar{u}_k\|^2 + 4\|\nabla f(\bar{z}_k)\|^2 + 4L^2\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{2\sigma^2}{n}\right). \tag{21}$$

Proof. By Lemma 1,

$$\bar{y}_{k+1} - \bar{x}_{k+1} = (1 - a_{k+1})\big((\bar{y}_k - \bar{x}_k) - \eta(\beta_1\bar{u}_k + \bar{g}_k)\big).$$

Young's inequality with $\tau = a_{k+1}/(1 - a_{k+1})$ gives

$$\|\bar{y}_{k+1} - \bar{x}_{k+1}\|^2 \le (1 - a_{k+1})\|\bar{y}_k - \bar{x}_k\|^2 + \frac{(1-a_{k+1})^2}{a_{k+1}}\eta^2\|\beta_1\bar{u}_k + \bar{g}_k\|^2.$$

Since $a_{k+1} \ge a_{\min} > 0$ and $t \mapsto (1-t)^2/t$ is decreasing on $(0,1)$,

$$1 - a_{k+1} \le 1 - a_{\min}, \qquad \frac{(1-a_{k+1})^2}{a_{k+1}} \le \frac{(1-a_{\min})^2}{a_{\min}}.$$

Taking $\mathbb{E}_k[\cdot]$ and applying Lemma 2 gives (21). □

Lemma 6. *For every $k \ge 0$,*

$$\mathbb{E}_k\|\bar{u}_{k+1}\|^2 \le \frac{1+\alpha_u^2}{2}\|\bar{u}_k\|^2 + 2\beta_2^2\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\|\nabla f(\bar{z}_k)\|^2 + 2L^2\beta_2^2\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \beta_2^2\frac{\sigma^2}{n}. \tag{22}$$

Proof. By Lemma 1 and $\mathbb{E}_k[\bar{g}_k] = \frac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)$,

$$\begin{aligned}\mathbb{E}_k\|\bar{u}_{k+1}\|^2 &= \alpha_u^2\|\bar{u}_k\|^2 + 2\alpha_u\beta_2\left\langle \bar{u}_k, \tfrac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)\right\rangle + \beta_2^2\mathbb{E}_k\|\bar{g}_k\|^2 \\ &\le \frac{1+\alpha_u^2}{2}\|\bar{u}_k\|^2 + \frac{2\alpha_u^2\beta_2^2}{1-\alpha_u^2}\left\|\tfrac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)\right\|^2 + \beta_2^2\mathbb{E}_k\|\bar{g}_k\|^2,\end{aligned} \tag{23}$$

where we apply Young's inequality with parameter $(1-\alpha_u^2)/2$ to the cross term. As in the proof of Lemma 2,

$$\left\|\tfrac{1}{n}\sum_{i=1}^n \nabla f_i(z_i^k)\right\|^2 \le 2\|\nabla f(\bar{z}_k)\|^2 + 2L^2\left(\tfrac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right).$$

Combining this bound with (15) in (23) gives (22). □

We will use the following lemma several times. It bounds the sum of a sequence satisfying a contractive recursion. We apply it to the expectations of $\|\bar{y}_k - \bar{x}_k\|^2$, $\|\bar{u}_k\|^2$, and later to the expected value of the consensus Lyapunov function.

Lemma 7. *Let $\{s_k\}_{k\ge0}$ and $\{r_k\}_{k\ge0}$ be nonnegative sequences such that $s_0 = 0$ and*

$$s_{k+1} \le q s_k + r_k, \qquad q \in [0,1).$$

Then, for every $T \ge 1$,

$$\sum_{k=0}^{T-1} s_k \le \frac{1}{1-q}\sum_{k=0}^{T-1} r_k. \tag{24}$$

Proof. Summing $s_{k+1} \le q s_k + r_k$ over $k = 0, \dots, T-1$ and using $s_0 = 0$ gives

$$(1-q)\sum_{k=0}^{T-1} s_k + s_T \le \sum_{k=0}^{T-1} r_k.$$

Dropping $s_T \ge 0$ and dividing by $1-q$ yields (24). □

Applying Lemma 7 to the two preceding inequalities and substituting the resulting bounds into Lemma 4 gives the following reduction.

Proposition 1. *Assume that*

$$\tilde{\eta} = \frac{\eta}{1-\beta_1} \le \frac{1}{8L}.$$

Let K_g, K_z, K_σ be the constants defined below. If $K_g > 0$, then for every $T \ge 1$,

$$\frac{1}{T}\sum_{k=0}^{T-1} \mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2 \le \frac{f(\bar{x}_0) - f^\star}{K_g\, T} + \frac{K_z}{K_g}\cdot\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{K_\sigma}{K_g}\frac{\sigma^2}{n}, \tag{25}$$

where

$$\begin{aligned}
K_g \coloneqq{}& \frac{\tilde{\eta}}{8} - 3\tilde{\eta}L^2(1-b_{\min})^2\left[\frac{4(1-a_{\min})^2\eta^2}{a_{\min}^2} + \frac{8\beta_1^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\right] \\
&- 3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\frac{4\beta_2^2}{1-\alpha_u^2}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right), \qquad (26)\\
K_z \coloneqq{}& 3\tilde{\eta}L^2(1-b_{\min})^2\left[\frac{4L^2(1-a_{\min})^2\eta^2}{a_{\min}^2} + \frac{8\beta_1^2L^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\right] \\
&+ 3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2 L^2\frac{4\beta_2^2}{1-\alpha_u^2}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right) \\
&+ \tilde{\eta}L^2 + L^3\tilde{\eta}^2, \qquad (27)\\
K_\sigma \coloneqq{}& 3\tilde{\eta}L^2(1-b_{\min})^2\left[\frac{2(1-a_{\min})^2\eta^2}{a_{\min}^2} + \frac{4\beta_1^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\right] \\
&+ 3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\frac{2\beta_2^2}{1-\alpha_u^2} \\
&+ \frac{L\tilde{\eta}^2}{2}. \qquad (28)
\end{aligned}$$

Proof. Applying Lemma 7 to Lemma 6 with $s_k = \mathbb{E}\left\|\bar{u}_k\right\|^2$, $q = (1+\alpha_u^2)/2$, and $1-q = (1-\alpha_u^2)/2$, and using $\bar{u}_0 = 0$, gives

$$\begin{aligned}
\sum_{k=0}^{T-1}\mathbb{E}\left\|\bar{u}_k\right\|^2 \le{}& \frac{4\beta_2^2}{1-\alpha_u^2}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2 \\
&+ \frac{4L^2\beta_2^2}{1-\alpha_u^2}\left(1 + \frac{2\alpha_u^2}{1-\alpha_u^2}\right)\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{2\beta_2^2}{1-\alpha_u^2}\,T\frac{\sigma^2}{n}. \qquad (29)
\end{aligned}$$

Similarly, applying Lemma 7 to Lemma 5 with $s_k = \mathbb{E}\|\bar{y}_k - \bar{x}_k\|^2$, $q = 1 - a_{\min}$, and using $\bar{y}_0 = \bar{x}_0$, yields

$$\begin{aligned}
\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k - \bar{x}_k\|^2 \le{}& \frac{2\beta_1^2(1-a_{\min})^2\eta^2}{a_{\min}^2}\sum_{k=0}^{T-1}\mathbb{E}\left\|\bar{u}_k\right\|^2 + \frac{4(1-a_{\min})^2\eta^2}{a_{\min}^2}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2 \\
&+ \frac{4L^2(1-a_{\min})^2\eta^2}{a_{\min}^2}\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + \frac{2(1-a_{\min})^2\eta^2}{a_{\min}^2}\,T\frac{\sigma^2}{n}.
\end{aligned} \tag{30}$$

Substituting (29) into (30) gives

$$\begin{aligned}
&\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2\\
&\le\left[\frac{4(1-a_{\min})^2\eta^2}{a_{\min}^2}+\frac{8\beta_1^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\left(1+\frac{2\alpha_u^2}{1-\alpha_u^2}\right)\right]\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2\\
&\quad+\left[\frac{4L^2(1-a_{\min})^2\eta^2}{a_{\min}^2}+\frac{8\beta_1^2L^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\left(1+\frac{2\alpha_u^2}{1-\alpha_u^2}\right)\right]\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\right)\\
&\quad+\left[\frac{2(1-a_{\min})^2\eta^2}{a_{\min}^2}+\frac{4\beta_1^2(1-a_{\min})^2\eta^2\beta_2^2}{a_{\min}^2(1-\alpha_u^2)}\right]T\frac{\sigma^2}{n}. \qquad (31)
\end{aligned}$$

Taking total expectations in (18), summing over $k=0,\dots,T-1$, and using $\hat{x}_0=\bar{x}_0$ and $f(\hat{x}_T)\ge f^\star$ gives

$$\begin{aligned}
\frac{\tilde{\eta}}{8}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2 &\le f(\bar{x}_0)-f^\star+3\tilde{\eta}L^2(1-b_{\min})^2\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2\\
&\quad+3\tilde{\eta}L^2\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2\sum_{k=0}^{T-1}\mathbb{E}\left\|\bar{u}_k\right\|^2+\left(\tilde{\eta}L^2+L^3\tilde{\eta}^2\right)\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\right)\\
&\quad+\frac{L\tilde{\eta}^2}{2}T\frac{\sigma^2}{n}. \qquad (32)
\end{aligned}$$

Substituting (29) and (31) into (32), then collecting the coefficients according to (26)–(28), gives

$$K_g\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2\le f(\bar{x}_0)-f^\star+K_z\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\right)+K_\sigma T\frac{\sigma^2}{n}. \qquad (33)$$

Since $K_g>0$, dividing (33) by K_gT proves (25). □

We next give convenient sufficient conditions under which K_g is positive and the ratios involving K_z and K_σ can be bounded by simpler quantities.

Lemma 8. *Under the assumptions of Proposition 1, suppose*

$$\eta^2\le\frac{1}{192\,L^2\left[\dfrac{(1-b_{\min})^2(1-a_{\min})^2}{a_{\min}^2}\left(1+\dfrac{2\beta_1^2(1+\alpha_u^2)}{(1-\beta_1)^2(1+\alpha_u)^2}\right)+\dfrac{\beta_1^2(1+\alpha_u^2)}{\beta_2^2(1-\beta_1)^4(1+\alpha_u)^2}\right]}. \qquad (34)$$

Then $K_g\ge\tilde{\eta}/16$. In particular, the simpler condition

$$\eta^2\le\frac{a_{\min}^2\beta_2^2(1-\beta_1)^4}{192L^2}$$

is sufficient.

Proof. Using

$$1-\alpha_u^2=\beta_2(1-\beta_1)(1+\alpha_u),\qquad 1+\frac{2\alpha_u^2}{1-\alpha_u^2}=\frac{1+\alpha_u^2}{1-\alpha_u^2},$$

we have

$$\frac{4\beta_2^2}{1-\alpha_u^2}\left(1+\frac{2\alpha_u^2}{1-\alpha_u^2}\right)=\frac{4(1+\alpha_u^2)}{(1-\beta_1)^2(1+\alpha_u)^2}. \qquad (35)$$

Substituting (35) into (26) gives

$$\begin{aligned}
\frac{\tilde{\eta}}{8}-K_g=12\tilde{\eta}L^2\eta^2\Bigg[&\frac{(1-b_{\min})^2(1-a_{\min})^2}{a_{\min}^2}\left(1+\frac{2\beta_1^2(1+\alpha_u^2)}{(1-\beta_1)^2(1+\alpha_u)^2}\right)\\
&+\frac{\beta_1^2(1+\alpha_u^2)}{\beta_2^2(1-\beta_1)^4(1+\alpha_u)^2}\Bigg]. \qquad (36)
\end{aligned}$$

Thus (34) implies

$$\frac{\tilde{\eta}}{8} - K_g \le \frac{\tilde{\eta}}{16},$$

and therefore $K_g \ge \tilde{\eta}/16$.

It remains to justify the simpler condition. Let B denote the bracket in (34). Since

$$(1-b_{\min})^2 \le 1, \qquad \frac{1+\alpha_u^2}{(1+\alpha_u)^2} \le 1, \qquad \beta_2 \le 1,$$

we obtain the chain

$$\begin{aligned} a_{\min}^2\beta_2^2(1-\beta_1)^4 B &\le (1-a_{\min})^2\beta_2^2(1-\beta_1)^4 + 2\beta_1^2(1-a_{\min})^2\beta_2^2(1-\beta_1)^2 + a_{\min}^2\beta_1^2 \\ &\le (1-a_{\min})^2(1-\beta_1)^2\big((1-\beta_1)^2 + 2\beta_1^2\big) + a_{\min}^2\beta_1^2 \\ &\le (1-a_{\min})^2 + a_{\min}^2 \le 1. \end{aligned} \tag{37}$$

Here the third inequality uses

$$(1-\beta_1)^2\big((1-\beta_1)^2 + 2\beta_1^2\big) \le 1, \qquad \beta_1^2 \le 1,$$

for $\beta_1 \in [0,1]$. Hence

$$B \le \frac{1}{a_{\min}^2\beta_2^2(1-\beta_1)^4}.$$

Therefore,

$$\eta^2 \le \frac{a_{\min}^2\beta_2^2(1-\beta_1)^4}{192L^2}$$

implies (34), completing the proof. □

Under the same condition, the next lemma gives simple upper bounds on K_z and K_σ.

Lemma 9. *Under the assumptions of Proposition 1 and the learning-rate condition* (34)*,*

$$K_z \le 2\,\tilde{\eta}L^2, \tag{38}$$

$$K_\sigma \le \left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\tilde{\eta}^2 L. \tag{39}$$

Proof. Comparing the definitions of K_z and K_g in (27) and (26), we obtain

$$K_z = L^2\left(\frac{\tilde{\eta}}{8} - K_g\right) + \tilde{\eta}L^2(1+L\tilde{\eta}).$$

Thus,

$$\begin{aligned} K_z &\overset{(a)}{\le} \frac{\tilde{\eta}L^2}{16} + \tilde{\eta}L^2(1+L\tilde{\eta}) \\ &\overset{(b)}{\le} \frac{\tilde{\eta}L^2}{16} + \frac{9}{8}\tilde{\eta}L^2 \le 2\tilde{\eta}L^2. \end{aligned} \tag{40}$$

Here, (a) uses Lemma 8, namely $K_g \ge \tilde{\eta}/16$, and (b) uses $L\tilde{\eta} \le 1/8$. This proves (38).

For K_σ, using $(1-b_{\min})^2 \le 1$, $(1-a_{\min})^2 \le 1$, $1-\alpha_u^2 \ge \beta_2(1-\beta_1)$, and

$$\left(\frac{\eta\beta_1}{\beta_2(1-\beta_1)}\right)^2 = \frac{\tilde{\eta}^2\beta_1^2}{\beta_2^2},$$

we have

$$\begin{aligned} K_\sigma &\overset{(c)}{\le} 3\tilde{\eta}L^2\left(\frac{2\eta^2}{a_{\min}^2} + \frac{4\beta_1^2\beta_2\eta^2}{a_{\min}^2(1-\beta_1)}\right) + 3\tilde{\eta}L^2\frac{\tilde{\eta}^2\beta_1^2}{\beta_2^2}\cdot\frac{2\beta_2^2}{\beta_2(1-\beta_1)} + \frac{L\tilde{\eta}^2}{2} \\ &\overset{(d)}{\le} \frac{18\tilde{\eta}^3L^2}{a_{\min}^2} + \frac{6\beta_1^2\tilde{\eta}^3L^2}{\beta_2(1-\beta_1)} + \frac{L\tilde{\eta}^2}{2} \\ &\overset{(e)}{\le} \frac{9\tilde{\eta}^2L}{4a_{\min}^2} + \frac{3\beta_1^2\tilde{\eta}^2L}{4\beta_2(1-\beta_1)} + \frac{L\tilde{\eta}^2}{2}. \end{aligned} \tag{41}$$

Here, (c) uses the elementary bounds stated above. For (d), we use

$$\frac{2\eta^2}{a_{\min}^2} + \frac{4\beta_1^2\beta_2\eta^2}{a_{\min}^2(1-\beta_1)} \le \frac{6\tilde{\eta}^2}{a_{\min}^2},$$

which follows from

$$2(1-\beta_1)^2 + 4\beta_1^2\beta_2(1-\beta_1) \le 6.$$

Finally, (e) uses $L\tilde{\eta} \le 1/8$. This proves (39). □

Remark 8. Under the basic learning-rate condition (11) and the sufficient condition (34), Lemma 8 and Lemma 9 yield

$$K_g \ge \frac{\tilde{\eta}}{16}, \qquad K_z \le 2\,\tilde{\eta}L^2, \qquad K_\sigma \le \left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\tilde{\eta}^2 L.$$

Consequently,

$$\frac{K_z}{K_g} \le 32L^2, \qquad \frac{K_\sigma}{K_g} \le 16\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\tilde{\eta}L.$$

These bounds are used in the convergence results below.

Proposition 1 shows that it remains to control the expected consensus error of Z_k, averaged over iterations. In particular, an $O(\eta^2)$ bound on

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\|Z_k - \bar{Z}_k\|_F^2\right)$$

gives an $O(1/\sqrt{T})$ bound on the iteration average of $\mathbb{E}\|\nabla f(\bar{z}_k)\|^2$ when $\eta = \Theta(1/\sqrt{T})$. We now prove such a consensus-error bound.

B.2 Step 2: consensus error bound

We now control the consensus error of Z_k. Since Z_k is formed from X_k and Y_k, we first bound the consensus errors of Y_k, U_k, and X_k, and then combine them to control Z_k. For the momentum variable U_k, we use the scaled consensus error

$$V_k \coloneqq \eta^2\left(\frac{1}{n}\|U_k - \bar{U}_k\|_F^2\right).$$

These bounds are then combined through a weighted Lyapunov function.

Remark 9. By Assumption 6,

$$\frac{1}{n}\left\|X_0 - \bar{X}_0\right\|_F^2 = \frac{1}{n}\left\|Y_0 - \bar{Y}_0\right\|_F^2 = \eta^2\frac{1}{n}\left\|U_0 - \bar{U}_0\right\|_F^2 = 0.$$

Thus, the initial consensus errors are zero, as required by Lemma 7.

We first bound the deviations of the local stochastic gradients from their network average in terms of the consensus error of Z_k, the gradient variance, and the data heterogeneity.

Lemma 10. *For every $k \ge 0$,*

$$\mathbb{E}_k\left(\frac{1}{n}\left\|G_k - \bar{G}_k\right\|_F^2\right) \le 4L^2\left(\frac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\right) + 4\zeta^2 + 2\sigma^2. \tag{42}$$

Proof. Define $H_k \coloneqq [\nabla f_1(z_1^k), \dots, \nabla f_n(z_n^k)]$. Since $\bar{G}_k = G_k J$ and $\bar{H}_k = H_k J$, we have

$$G_k - \bar{G}_k = G_k(I-J) = (G_k - H_k)(I-J) + (H_k - \bar{H}_k).$$

Taking the squared Frobenius norm and then the conditional expectation gives

$$\begin{aligned}
\mathbb{E}_k\|G_k - \bar{G}_k\|_F^2 &\le 2\,\mathbb{E}_k\|(G_k - H_k)(I-J)\|_F^2 + 2\,\|H_k - \bar{H}_k\|_F^2 \\
&\le 2\,\mathbb{E}_k\|G_k - H_k\|_F^2 + 2\,\|H_k - \bar{H}_k\|_F^2 \\
&\le 2n\sigma^2 + 2\,\|H_k - \bar{H}_k\|_F^2 \\
&\le 2n\sigma^2 + 2\sum_{i=1}^n \|\nabla f_i(z_i^k) - \nabla f(\bar{z}_k)\|^2 \\
&\le 2n\sigma^2 + 4\sum_{i=1}^n \|\nabla f_i(z_i^k) - \nabla f_i(\bar{z}_k)\|^2 + 4\sum_{i=1}^n \|\nabla f_i(\bar{z}_k) - \nabla f(\bar{z}_k)\|^2 \\
&\le 2n\sigma^2 + 4L^2\,\|Z_k - \bar{Z}_k\|_F^2 + 4n\zeta^2. \qquad (43)
\end{aligned}$$

The second inequality uses $\|I-J\|_2 \le 1$, and the third uses Assumption 2. The fourth inequality uses the fact that the column average minimizes the sum of squared distances. The last two inequalities use $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$, Assumption 1, and Assumption 3. Dividing by n gives (42). □

The next bound tracks how the Y-update affects the consensus error.

Lemma 11. *Under Assumption 4, for every $k \ge 0$,*

$$\begin{aligned}
&\mathbb{E}_k\left(\frac{1}{n}\left\|Y_{k+1} - \bar{Y}_{k+1}\right\|_F^2\right) \\
\le\ &\left(1 - \frac{\rho(2-\rho)}{2}\right)\left(\frac{1}{n}\left\|Y_k - \bar{Y}_k\right\|_F^2\right) + \frac{4\beta_1^2}{\rho(2-\rho)}\,\eta^2\left(\frac{1}{n}\left\|U_k - \bar{U}_k\right\|_F^2\right) \\
&+ \frac{4\eta^2}{\rho(2-\rho)}\,\mathbb{E}_k\left(\frac{1}{n}\left\|G_k - \bar{G}_k\right\|_F^2\right). \qquad (44)
\end{aligned}$$

Proof. From the Y-update (3) and $W\mathbf{1} = \mathbf{1}$, we have

$$Y_{k+1} - \bar{Y}_{k+1} = \big((Y_k - \bar{Y}_k) - \eta(\beta_1(U_k - \bar{U}_k) + (G_k - \bar{G}_k))\big)W. \qquad (45)$$

Applying Young's inequality

$$\|p+q\|_F^2 \le (1+\tau)\|p\|_F^2 + \left(1 + \frac{1}{\tau}\right)\|q\|_F^2$$

to (45) with

$$\tau = \frac{\rho(2-\rho)}{2(1-\rho)^2},$$

we obtain

$$\begin{aligned}
\|Y_{k+1} - \bar{Y}_{k+1}\|_F^2 &\le (1+\tau)\|(Y_k - \bar{Y}_k)W\|_F^2 \\
&\quad + \left(1 + \frac{1}{\tau}\right)\eta^2\|(\beta_1(U_k - \bar{U}_k) + (G_k - \bar{G}_k))W\|_F^2 \\
&\overset{(a)}{\le} (1+\tau)(1-\rho)^2\|Y_k - \bar{Y}_k\|_F^2 \\
&\quad + \left(1 + \frac{1}{\tau}\right)\eta^2\|\beta_1(U_k - \bar{U}_k) + (G_k - \bar{G}_k)\|_F^2 \\
&\overset{(b)}{\le} \left(1 - \frac{\rho(2-\rho)}{2}\right)\|Y_k - \bar{Y}_k\|_F^2 + \frac{2\eta^2}{\rho(2-\rho)}\|\beta_1(U_k - \bar{U}_k) + (G_k - \bar{G}_k)\|_F^2 \\
&\overset{(c)}{\le} \left(1 - \frac{\rho(2-\rho)}{2}\right)\|Y_k - \bar{Y}_k\|_F^2 + \frac{4\beta_1^2\eta^2}{\rho(2-\rho)}\|U_k - \bar{U}_k\|_F^2 \\
&\quad + \frac{4\eta^2}{\rho(2-\rho)}\|G_k - \bar{G}_k\|_F^2. \qquad (46)
\end{aligned}$$

Here, (a) uses Remark 7 applied to $A = Y_k - \bar{Y}_k$, which is centered, for the first term, and $\|W\|_2 \le 1$ from Remark 6 for the second term. (b) follows from the choice of τ, since

$$(1+\tau)(1-\rho)^2 = 1 - \frac{\rho(2-\rho)}{2}, \qquad 1 + \frac{1}{\tau} \le \frac{2}{\rho(2-\rho)}.$$

(c) uses

$$\|\beta_1(U_k - \bar{U}_k) + (G_k - \bar{G}_k)\|_F^2 \le 2\beta_1^2 \|U_k - \bar{U}_k\|_F^2 + 2\|G_k - \bar{G}_k\|_F^2.$$

Dividing (46) by n and taking $\mathbb{E}_k[\cdot]$ gives (44). □

We next control the consensus error of the momentum variable.

Lemma 12. *Define*

$$V_k \coloneqq \eta^2 \left(\frac{1}{n} \left\| U_k - \bar{U}_k \right\|_F^2 \right), \qquad \alpha_{u,\rho} \coloneqq (1-\beta_2) + \beta_2\beta_1(1-\rho).$$

Then, under Assumption 4, for every $k \ge 0$,

$$\begin{aligned} \mathbb{E}_k V_{k+1} \le \frac{1+\alpha_{u,\rho}^2}{2} V_k & \\ + \frac{8\left(1+\alpha_{u,\rho}^2\right)\beta_2^2}{1-\alpha_{u,\rho}^2} & \left(\frac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) \\ + \frac{2\left(1+\alpha_{u,\rho}^2\right)\beta_2^2}{1-\alpha_{u,\rho}^2} & \eta^2 \mathbb{E}_k \left(\frac{1}{n} \left\| G_k - \bar{G}_k \right\|_F^2 \right). \end{aligned} \tag{47}$$

Proof. Multiplying the U-update by η, we get

$$\eta U_{k+1} = (1-\beta_2)\eta U_k + \beta_2(Y_k - Y_{k+1}).$$

Using the Y-update (3), this becomes

$$\eta U_{k+1} = (1-\beta_2)\eta U_k + \beta_2 Y_k(I - W) + \beta_2\eta(\beta_1 U_k + G_k)W.$$

Since $W\mathbf{1} = \mathbf{1}$, subtracting the column average from each column gives

$$\begin{aligned} \eta(U_{k+1} - \bar{U}_{k+1}) = \eta(U_k - \bar{U}_k)\big((1-\beta_2)I + \beta_2\beta_1 W\big) & \\ + \beta_2(Y_k - \bar{Y}_k)(I - W) + \beta_2\eta(G_k - \bar{G}_k)W. & \end{aligned} \tag{48}$$

For the first term in (48), Assumption 4 gives

$$\begin{aligned} &\left\| \eta(U_k - \bar{U}_k)\big((1-\beta_2)I + \beta_2\beta_1 W\big) \right\|_F \\ &\quad \le \eta(1-\beta_2)\|U_k - \bar{U}_k\|_F + \eta\beta_2\beta_1\|(U_k - \bar{U}_k)W\|_F \\ &\quad \le \eta\big((1-\beta_2) + \beta_2\beta_1(1-\rho)\big)\|U_k - \bar{U}_k\|_F \\ &\quad = \alpha_{u,\rho}\eta\|U_k - \bar{U}_k\|_F. \end{aligned} \tag{49}$$

The first inequality uses the triangle inequality, and the second uses Remark 7. Applying Young's inequality in the form

$$\|a+b+c\|_F^2 \le (1+\gamma)\|a\|_F^2 + 2\left(1 + \frac{1}{\gamma}\right)\left(\|b\|_F^2 + \|c\|_F^2\right)$$

to (48), with

$$\gamma = \frac{1-\alpha_{u,\rho}^2}{2\alpha_{u,\rho}^2},$$

and using (49), we get

$$\begin{aligned}\eta^2\|U_{k+1}-\bar{U}_{k+1}\|_F^2 &\le \frac{1+\alpha_{u,\rho}^2}{2}\,\eta^2\|U_k-\bar{U}_k\|_F^2\\ &\quad+\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\|(Y_k-\bar{Y}_k)(I-W)\|_F^2\\ &\quad+\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\eta^2\|(G_k-\bar{G}_k)W\|_F^2\\ &\le \frac{1+\alpha_{u,\rho}^2}{2}\,\eta^2\|U_k-\bar{U}_k\|_F^2\\ &\quad+\frac{8(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\|Y_k-\bar{Y}_k\|_F^2\\ &\quad+\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\eta^2\|G_k-\bar{G}_k\|_F^2. \end{aligned}\tag{50}$$

The last inequality uses $\|I-W\|_2\le 2$ and $\|W\|_2\le 1$. Dividing (50) by n and taking $\mathbb{E}_k[\cdot]$ gives (47). □

The consensus error of the output sequence X is bounded in terms of the consensus errors of X_k and Y_{k+1}.

Lemma 13. *Under Assumption 4, for every $k\ge 0$,*

$$\begin{aligned}\mathbb{E}_k\left(\frac{1}{n}\left\|X_{k+1}-\bar{X}_{k+1}\right\|_F^2\right) \le (1-\rho)^2\Bigg[&(1-a_{k+1})\left(\frac{1}{n}\left\|X_k-\bar{X}_k\right\|_F^2\right)\\ &+a_{k+1}\mathbb{E}_k\left(\frac{1}{n}\left\|Y_{k+1}-\bar{Y}_{k+1}\right\|_F^2\right)\Bigg].\end{aligned}\tag{51}$$

Proof. From the X-update (5) and $W\mathbf{1}=\mathbf{1}$,

$$X_{k+1}-\bar{X}_{k+1}=\big((1-a_{k+1})(X_k-\bar{X}_k)+a_{k+1}(Y_{k+1}-\bar{Y}_{k+1})\big)W.\tag{52}$$

Moreover,

$$\big((1-a_{k+1})(X_k-\bar{X}_k)+a_{k+1}(Y_{k+1}-\bar{Y}_{k+1})\big)\mathbf{1}=0.$$

Therefore, by Remark 7,

$$\begin{aligned}\|X_{k+1}-\bar{X}_{k+1}\|_F^2 &\le (1-\rho)^2\big\|(1-a_{k+1})(X_k-\bar{X}_k)+a_{k+1}(Y_{k+1}-\bar{Y}_{k+1})\big\|_F^2\\ &\le (1-\rho)^2\Big[(1-a_{k+1})\|X_k-\bar{X}_k\|_F^2+a_{k+1}\|Y_{k+1}-\bar{Y}_{k+1}\|_F^2\Big].\end{aligned}\tag{53}$$

The second inequality uses convexity of $\|\cdot\|_F^2$ and $a_{k+1}\in[0,1]$. Dividing by n and taking $\mathbb{E}_k[\cdot]$ proves (51). □

The definition of Z_k as a weighted average gives the following consensus error bound.

Lemma 14. *For every $k\ge 0$,*

$$\frac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\le 2\left(\frac{1}{n}\left\|X_k-\bar{X}_k\right\|_F^2\right)+2\left(\frac{1}{n}\left\|Y_k-\bar{Y}_k\right\|_F^2\right).\tag{54}$$

Proof. From the Z-update (2),

$$Z_k-\bar{Z}_k=b_{k+1}(Y_k-\bar{Y}_k)+(1-b_{k+1})(X_k-\bar{X}_k).$$

Therefore,

$$\begin{aligned}\left\|Z_k-\bar{Z}_k\right\|_F^2 &\le 2b_{k+1}^2\left\|Y_k-\bar{Y}_k\right\|_F^2+2(1-b_{k+1})^2\left\|X_k-\bar{X}_k\right\|_F^2\\ &\le 2\left\|Y_k-\bar{Y}_k\right\|_F^2+2\left\|X_k-\bar{X}_k\right\|_F^2.\end{aligned}\tag{55}$$

The first inequality uses $\|p+q\|_F^2\le 2\|p\|_F^2+2\|q\|_F^2$, and the second uses $b_{k+1}\in[0,1]$. Dividing by n proves (54). □

B.3 Weighted Lyapunov bound

We combine the preceding consensus-error inequalities into a single Lyapunov function. The weights are chosen so that the terms involving X_k, Y_k, and V_k contract jointly in conditional expectation, up to terms due to gradient variance and data heterogeneity. For weights $\lambda_X, \lambda_Y, \lambda_V > 0$, define

$$\Psi_k \coloneqq \lambda_X \left(\tfrac{1}{n} \left\| X_k - \bar{X}_k \right\|_F^2 \right) + \lambda_Y \left(\tfrac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) + \lambda_V V_k.$$

Lemma 15 gives a one-step inequality for Ψ_k. Proposition 2 then gives explicit weights for which

$$\mathbb{E}_k \Psi_{k+1} \le (1-\chi)\Psi_k + C_\Psi,$$

and Proposition 3 sums this inequality to obtain the bound on the expected consensus error, averaged over iterations.

Lemma 15. *Under Assumption 4 with $\rho \in (0,1)$, for any $\lambda_X, \lambda_Y, \lambda_V > 0$,*

$$\mathbb{E}_k \Psi_{k+1} \le \theta_X \left(\tfrac{1}{n} \left\| X_k - \bar{X}_k \right\|_F^2 \right) + \theta_Y \left(\tfrac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) + \theta_V V_k + C_\Psi, \tag{56}$$

where

$$\theta_X \coloneqq \lambda_X (1-\rho)^2 + 8L^2\eta^2 \left[\left(\lambda_X (1-\rho)^2 + \lambda_Y \right) \frac{4}{\rho(2-\rho)} + \lambda_V \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \right], \tag{57}$$

$$\begin{aligned} \theta_Y \coloneqq{}& \lambda_X (1-\rho)^2 \left(1 - \frac{\rho(2-\rho)}{2} \right) + \lambda_Y \left(1 - \frac{\rho(2-\rho)}{2} \right) + \lambda_V \frac{8(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \\ &+ 8L^2\eta^2 \left[\lambda_X (1-\rho)^2 \frac{4}{\rho(2-\rho)} + \lambda_Y \frac{4}{\rho(2-\rho)} + \lambda_V \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \right], \end{aligned} \tag{58}$$

$$\theta_V \coloneqq \left(\lambda_X (1-\rho)^2 + \lambda_Y \right) \frac{4\beta_1^2}{\rho(2-\rho)} + \lambda_V \frac{1+\alpha_{u,\rho}^2}{2}, \tag{59}$$

$$C_\Psi \coloneqq \left[\left(\lambda_X (1-\rho)^2 + \lambda_Y \right) \frac{4}{\rho(2-\rho)} + \lambda_V \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \right] \eta^2 (4\zeta^2 + 2\sigma^2). \tag{60}$$

In particular, if for some $\chi \in (0,1)$,

$$\theta_X \le (1-\chi)\lambda_X, \qquad \theta_Y \le (1-\chi)\lambda_Y, \qquad \theta_V \le (1-\chi)\lambda_V, \tag{61}$$

then

$$\mathbb{E}_k \Psi_{k+1} \le (1-\chi)\Psi_k + C_\Psi.$$

Proof. Combining Lemma 10 with Lemma 14 gives

$$\mathbb{E}_k \left(\tfrac{1}{n} \left\| G_k - \bar{G}_k \right\|_F^2 \right) \le 8L^2 \left(\tfrac{1}{n} \left\| X_k - \bar{X}_k \right\|_F^2 \right) + 8L^2 \left(\tfrac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) + 4\zeta^2 + 2\sigma^2. \tag{62}$$

Substituting (62) into Lemma 11 gives

$$\begin{aligned} \mathbb{E}_k \left(\tfrac{1}{n} \left\| Y_{k+1} - \bar{Y}_{k+1} \right\|_F^2 \right) \le{}& 8L^2\eta^2 \frac{4}{\rho(2-\rho)} \left(\tfrac{1}{n} \left\| X_k - \bar{X}_k \right\|_F^2 \right) \\ &+ \left(1 - \frac{\rho(2-\rho)}{2} + 8L^2\eta^2 \frac{4}{\rho(2-\rho)} \right) \left(\tfrac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) \\ &+ \frac{4\beta_1^2}{\rho(2-\rho)} V_k + \frac{4}{\rho(2-\rho)} \eta^2 (4\zeta^2 + 2\sigma^2). \end{aligned} \tag{63}$$

Similarly, Lemma 12 and (62) give

$$\begin{aligned} \mathbb{E}_k V_{k+1} \le{}& 8L^2\eta^2 \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \left(\tfrac{1}{n} \left\| X_k - \bar{X}_k \right\|_F^2 \right) \\ &+ \left[\frac{8(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} + 8L^2\eta^2 \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \right] \left(\tfrac{1}{n} \left\| Y_k - \bar{Y}_k \right\|_F^2 \right) \\ &+ \frac{1+\alpha_{u,\rho}^2}{2} V_k + \frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} \eta^2 (4\zeta^2 + 2\sigma^2). \end{aligned} \tag{64}$$

Lemma 13, combined with (63) and the bounds $1 - a_{k+1} \le 1$ and $a_{k+1} \le 1$, gives

$$\begin{aligned}\mathbb{E}_k\Big(\tfrac{1}{n}\left\|X_{k+1} - \bar{X}_{k+1}\right\|_F^2\Big) \le & \left((1-\rho)^2 + 8L^2\eta^2(1-\rho)^2\frac{4}{\rho(2-\rho)}\right)\left(\tfrac{1}{n}\left\|X_k - \bar{X}_k\right\|_F^2\right) \\ & + (1-\rho)^2\left(1 - \frac{\rho(2-\rho)}{2} + 8L^2\eta^2\frac{4}{\rho(2-\rho)}\right)\left(\tfrac{1}{n}\left\|Y_k - \bar{Y}_k\right\|_F^2\right) \\ & + (1-\rho)^2\frac{4\beta_1^2}{\rho(2-\rho)}V_k + (1-\rho)^2\frac{4}{\rho(2-\rho)}\eta^2(4\zeta^2 + 2\sigma^2). \quad (65)\end{aligned}$$

Multiplying (65), (63), and (64) by $\lambda_X, \lambda_Y, \lambda_V$, respectively, and summing gives (56) with the coefficients in (57)–(60).

If (61) holds, then the first three terms in (56) are bounded by $(1-\chi)\Psi_k$. Hence

$$\mathbb{E}_k\Psi_{k+1} \le (1-\chi)\Psi_k + C_\Psi,$$

which proves the claim. □

Proposition 2. *Under Assumption 4, let*

$$\delta_\rho := \rho(2-\rho), \qquad \mathcal{M} := \frac{\frac{4\beta_1^2}{\delta_\rho}\cdot\frac{8(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}}{1-\alpha_{u,\rho}^2}.$$

Suppose that

$$64\,\mathcal{M} \le \delta_\rho. \quad (66)$$

Choose

$$\lambda_X := 1, \quad (67)$$

$$\lambda_Y := \frac{16\left[(1-\rho)^2\left(1-\frac{\delta_\rho}{2}\right) + 8(1-\rho)^2\mathcal{M} + \frac{7\delta_\rho}{8}\right]}{\delta_\rho}, \quad (68)$$

$$\lambda_V := \frac{8\big((1-\rho)^2 + \lambda_Y\big)\frac{4\beta_1^2}{\delta_\rho}}{1-\alpha_{u,\rho}^2}. \quad (69)$$

Suppose further that

$$\eta^2 \le \min\left\{\frac{7\delta_\rho}{64L^2\left[\big((1-\rho)^2+\lambda_Y\big)\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right]},\ \frac{\lambda_Y\delta_\rho}{64L^2\left[\lambda_Y\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right]}\right\}. \quad (70)$$

Then, with

$$\chi := \frac{1}{8}\min\left\{\delta_\rho,\, 1-\alpha_{u,\rho}^2\right\},$$

the contraction conditions (61) *hold.*

Proof. We verify the three inequalities in (61).

First, we verify the condition for θ_V. Since $\lambda_X = 1$, the definition of θ_V gives

$$\begin{aligned}\theta_V &= \big((1-\rho)^2 + \lambda_Y\big)\frac{4\beta_1^2}{\delta_\rho} + \lambda_V\frac{1+\alpha_{u,\rho}^2}{2} \\ &= \frac{1-\alpha_{u,\rho}^2}{8}\lambda_V + \lambda_V\frac{1+\alpha_{u,\rho}^2}{2} \\ &= \lambda_V\left(\frac{1-\alpha_{u,\rho}^2}{8} + \frac{1+\alpha_{u,\rho}^2}{2}\right) \\ &= \lambda_V\left(\frac{5+3\alpha_{u,\rho}^2}{8}\right) \\ &= \lambda_V\left(1 - \frac{3(1-\alpha_{u,\rho}^2)}{8}\right). \quad (71)\end{aligned}$$

Here, the second equality uses the choice of λ_V in (69). Therefore, $\theta_V \le (1-\chi)\lambda_V$ whenever

$$\chi \le \frac{3(1-\alpha_{u,\rho}^2)}{8}. \tag{72}$$

Next, we verify the condition for θ_Y. Define

$$A \coloneqq (1-\rho)^2\left(1-\frac{\delta_\rho}{2}\right) + 8(1-\rho)^2\mathcal{M}.$$

By the choice of λ_V in (69),

$$\lambda_V \frac{8(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2} = 8\big((1-\rho)^2+\lambda_Y\big)\mathcal{M}.$$

Using this identity in the definition of θ_Y, we get

$$\begin{aligned}
\theta_Y &= A + \lambda_Y\left(1-\frac{\delta_\rho}{2}+8\mathcal{M}\right) + 8L^2\eta^2\left[(1-\rho)^2\frac{4}{\delta_\rho} + \lambda_Y\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right] \\
&\le A + \lambda_Y\left(1-\frac{\delta_\rho}{2}+8\mathcal{M}\right) + \frac{7\delta_\rho}{8} + \frac{\delta_\rho}{8}\lambda_Y \\
&= A + \frac{7\delta_\rho}{8} + \lambda_Y\left(1-\frac{3\delta_\rho}{8}+8\mathcal{M}\right) \\
&\le A + \frac{7\delta_\rho}{8} + \lambda_Y\left(1-\frac{\delta_\rho}{4}\right) \\
&= \frac{\delta_\rho}{16}\lambda_Y + \lambda_Y\left(1-\frac{\delta_\rho}{4}\right) \\
&= \lambda_Y\left(1-\frac{3\delta_\rho}{16}\right).
\end{aligned} \tag{73}$$

Here, the first inequality uses the learning-rate condition (70); the second inequality uses (66), namely $8\mathcal{M} \le \delta_\rho/8$; and the penultimate line uses the definition of λ_Y in (68). Therefore, $\theta_Y \le (1-\chi)\lambda_Y$ whenever

$$\chi \le \frac{3\delta_\rho}{16}. \tag{74}$$

Finally, we verify the condition for θ_X. Since $\lambda_X = 1$, the definition of θ_X gives

$$\begin{aligned}
\theta_X &= (1-\rho)^2 + 8L^2\eta^2\left[\big((1-\rho)^2+\lambda_Y\big)\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right] \\
&\le (1-\rho)^2 + \frac{7\delta_\rho}{8} \\
&= 1-\delta_\rho+\frac{7\delta_\rho}{8} \\
&= 1-\frac{\delta_\rho}{8}.
\end{aligned} \tag{75}$$

Here, the inequality uses the first part of the learning-rate condition (70), and the first equality after it uses $(1-\rho)^2 = 1-\delta_\rho$. Therefore, $\theta_X \le (1-\chi)\lambda_X$ whenever

$$\chi \le \frac{\delta_\rho}{8}. \tag{76}$$

It remains to check that our choice of χ satisfies the three requirements. Since

$$\chi = \frac{1}{8}\min\big\{\delta_\rho,\, 1-\alpha_{u,\rho}^2\big\},$$

we have

$$\chi \le \frac{1-\alpha_{u,\rho}^2}{8} \le \frac{3(1-\alpha_{u,\rho}^2)}{8}, \qquad \chi \le \frac{\delta_\rho}{8} \le \frac{3\delta_\rho}{16}.$$

Thus (72), (74), and (76) all hold. Moreover, $\delta_\rho > 0$ and $\alpha_{u,\rho} < 1$ because $\rho, \beta_1, \beta_2 \in (0,1)$, so $\chi > 0$. Hence (61) holds. □

Remark 10. The condition on the momentum parameters (66) is used only to obtain an explicit contraction factor for the weighted consensus Lyapunov function. It restricts the admissible momentum parameters as a function of the spectral gap ρ: as ρ decreases, the condition on the momentum parameters becomes more restrictive. For example, when β_2 is bounded away from zero, it suffices to take $\beta_1 \le c_\beta \rho$ with a sufficiently small constant $c_\beta > 0$.

Analogous conditions on the momentum parameters also appear in convergence analyses of quasi-global momentum methods [Lin et al., 2021]; for example, QG-DSGDm imposes a condition of the form $\beta/(1-\beta) \lesssim \rho$. These restrictions are sufficient conditions for the proof, not practical tuning rules. They arise from the particular Lyapunov weights and bounds from Young's inequality used to obtain a joint contraction.

In experiments, we tune β_1 and β_2 empirically. As is common for momentum methods, training can remain stable and effective outside the range covered by these sufficient conditions; in our experiments, the selected ring values of β_1 are 0.90 and 0.95.

Summing the Lyapunov inequality gives the following bound on the expected consensus error, averaged over iterations.

Proposition 3. *Suppose Assumptions 4 and 6 and the assumptions of Proposition 2 hold. Then, for every $k \ge 0$,*

$$\mathbb{E}_k \Psi_{k+1} \le (1-\chi)\Psi_k + C_\Psi, \tag{77}$$

and, for every $T \ge 1$,

$$\frac{1}{T}\sum_{k=0}^{T-1} \mathbb{E}\Big(\tfrac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2\Big) \le \frac{2C_\Psi}{\chi \min\{\lambda_X, \lambda_Y\}}. \tag{78}$$

Proof. By Proposition 2, the contraction conditions (61) hold with the stated χ. Substituting them into Lemma 15 gives (77).

Taking total expectation in (77) and applying Lemma 7 with

$$s_k = \mathbb{E}\Psi_k, \qquad q = 1-\chi, \qquad r_k = C_\Psi,$$

and using $\Psi_0 = 0$, yields

$$\sum_{k=0}^{T-1} \mathbb{E}\Psi_k \le \frac{C_\Psi}{\chi} T. \tag{79}$$

Here, $\Psi_0 = 0$ follows from Assumption 6 and the definition of Ψ_k.

By Lemma 14,

$$\tfrac{1}{n}\left\|Z_k - \bar{Z}_k\right\|_F^2 \le 2\left(\tfrac{1}{n}\left\|X_k - \bar{X}_k\right\|_F^2\right) + 2\left(\tfrac{1}{n}\left\|Y_k - \bar{Y}_k\right\|_F^2\right) \le \frac{2}{\min\{\lambda_X, \lambda_Y\}}\Psi_k.$$

Summing over $k = 0, \dots, T-1$, taking expectation, dividing by T, and using (79) gives (78). □

Remark 11. With the choices in Proposition 2, $\lambda_X, \lambda_Y, \lambda_V$ and χ depend only on ρ, β_1, β_2. Hence, from (60),

$$C_\Psi = O\big(\eta^2(\sigma^2 + \zeta^2)\big),$$

where the hidden constant depends only on ρ, β_1, β_2. Therefore, under the conditions of Proposition 3,

$$\frac{1}{T}\sum_{k=0}^{T-1} \mathbb{E}\left(\frac{1}{n}\|Z_k - \bar{Z}_k\|_F^2\right) = O\big(\eta^2(\sigma^2 + \zeta^2)\big).$$

Remark 12. Let η_{cons}^2 denote the right-hand side of (70). Assume the condition on the momentum parameters (66) holds. Let

$$\eta = \frac{c}{\sqrt{T}}, \qquad 0 < c \le c_0,$$

where $c_0 > 0$ is chosen so that

$$c_0^2 \le \min\left\{\frac{(1-\beta_1)^2}{64L^2},\ \frac{a_{\min}^2 \beta_2^2 (1-\beta_1)^4}{192L^2},\ \eta_{\mathrm{cons}}^2\right\}.$$

Then, for every $T \ge 1$, the choice $\eta = c/\sqrt{T}$ satisfies the basic learning-rate condition (11), the sufficient condition (34), and the consensus learning-rate condition (70).

B.4 Proof of Theorem 1

Proof. By Proposition 1 and (78),

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\,\|\nabla f(\bar{z}_k)\|^2 \le \frac{f(\bar{x}_0)-f^\star}{K_g T} + \frac{K_z}{K_g}\cdot\frac{2C_\Psi}{\chi\min\{\lambda_X,\lambda_Y\}} + \frac{K_\sigma}{K_g}\cdot\frac{\sigma^2}{n}.$$

Under (34), Lemmas 8 and 9 give

$$\frac{1}{K_g} \le \frac{16}{\tilde{\eta}}, \qquad \frac{K_z}{K_g} \le 32L^2, \qquad \frac{K_\sigma}{K_g} \le 16\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\tilde{\eta}L.$$

Substituting these bounds proves (6).

Now choose $\eta = c/\sqrt{T}$. Then

$$\tilde{\eta} = \frac{c}{(1-\beta_1)\sqrt{T}}, \qquad \eta^2 = \frac{c^2}{T}.$$

With these identities, the term involving the initial objective gap in (6) becomes

$$\frac{16(1-\beta_1)(f(\bar{x}_0)-f^\star)}{c\sqrt{T}},$$

the gradient-variance term becomes

$$\frac{16}{1-\beta_1}\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\frac{Lc\sigma^2}{n\sqrt{T}},$$

and the consensus-error term, by (60), becomes

$$\frac{64L^2c^2}{\chi\min\{\lambda_X,\lambda_Y\}\,T}\left[\big(\lambda_X(1-\rho)^2+\lambda_Y\big)\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right](4\zeta^2+2\sigma^2).$$

Collecting $1/\sqrt{T}$ terms into $C_{z,1}$ and $1/T$ terms into $C_{z,2}$ proves (7), where

$$C_{z,1} \coloneqq \frac{16(1-\beta_1)(f(\bar{x}_0)-f^\star)}{c} + \frac{16}{1-\beta_1}\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\frac{Lc\sigma^2}{n}, \tag{80}$$

$$C_{z,2} \coloneqq \frac{64L^2c^2}{\chi\min\{\lambda_X,\lambda_Y\}}\left[\big(\lambda_X(1-\rho)^2+\lambda_Y\big)\frac{4}{\delta_\rho} + \lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right](4\zeta^2+2\sigma^2). \tag{81}$$

□

Proof of Corollary 1. By (8), the choice

$$\eta = c\sqrt{\frac{n}{T}}$$

satisfies the learning-rate conditions used in Theorem 1. Moreover,

$$\tilde{\eta} = \frac{c}{1-\beta_1}\sqrt{\frac{n}{T}}, \qquad \eta^2 = \frac{c^2 n}{T}.$$

With these identities, the term involving the initial objective gap in (6) becomes

$$\frac{16(1-\beta_1)(f(\bar{x}_0)-f^\star)}{c\sqrt{nT}},$$

the gradient-variance term becomes

$$\frac{16}{1-\beta_1}\left(\frac{9}{4a_{\min}^2} + \frac{3\beta_1^2}{4\beta_2(1-\beta_1)} + \frac{1}{2}\right)\frac{cL\sigma^2}{\sqrt{nT}},$$

and, by (60), the consensus-error term becomes

$$\frac{64L^2c^2n}{\chi\min\{\lambda_X,\lambda_Y\}T}\left[\left(\lambda_X(1-\rho)^2+\lambda_Y\right)\frac{4}{\delta_\rho}+\lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right](4\zeta^2+2\sigma^2).$$

Collecting the first two terms into $D_1/\sqrt{nT}$ and the third term into $D_2 n/T$ gives (9), where

$$D_1 \coloneqq \frac{16(1-\beta_1)(f(\bar{x}_0)-f^\star)}{c}+\frac{16}{1-\beta_1}\left(\frac{9}{4a_{\min}^2}+\frac{3\beta_1^2}{4\beta_2(1-\beta_1)}+\frac{1}{2}\right)cL\sigma^2, \tag{82}$$

$$D_2 \coloneqq \frac{64L^2c^2}{\chi\min\{\lambda_X,\lambda_Y\}}\left[\left(\lambda_X(1-\rho)^2+\lambda_Y\right)\frac{4}{\delta_\rho}+\lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right](4\zeta^2+2\sigma^2). \tag{83}$$

□

C Proof of Theorem 2

We transfer the bound on the expected squared gradient norm, averaged over iterations, from $\bar{z}_k$ to the network-average output model $\bar{x}_k$. Since

$$\bar{z}_k-\bar{x}_k=b_{k+1}(\bar{y}_k-\bar{x}_k),$$

it remains to control the expected squared distance between $\bar{y}_k$ and $\bar{x}_k$, averaged over iterations.

For compactness, define

$$B_\Psi \coloneqq \left[\left(\lambda_X(1-\rho)^2+\lambda_Y\right)\frac{4}{\delta_\rho}+\lambda_V\frac{2(1+\alpha_{u,\rho}^2)\beta_2^2}{1-\alpha_{u,\rho}^2}\right](4\zeta^2+2\sigma^2).$$

Then $C_\Psi=\eta^2 B_\Psi$. Under the choice $\eta=c/\sqrt{T}$, Proposition 3 gives

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left(\frac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\right)\le\frac{C_{Z,2}}{T},\qquad C_{Z,2}\coloneqq\frac{2c^2B_\Psi}{\chi\min\{\lambda_X,\lambda_Y\}}.$$

The constants in (10) may be chosen as

$$C_{x,1}\coloneqq 2C_{z,1}, \tag{84}$$

$$C_{x,2}\coloneqq 2C_{z,2}+\frac{12L^2b_{\max}^2c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{\sigma^2}{n}, \tag{85}$$

$$C_{x,3}\coloneqq\frac{56L^2b_{\max}^2c^2}{a_{\min}^2(1-\beta_1)^2}\left(C_{z,1}+C_{z,2}\right)+\frac{56L^4b_{\max}^2c^2}{a_{\min}^2(1-\beta_1)^2}C_{Z,2}. \tag{86}$$

Proof. By $\bar{z}_k-\bar{x}_k=b_{k+1}(\bar{y}_k-\bar{x}_k)$, $\|p+q\|^2\le 2\|p\|^2+2\|q\|^2$, Assumption 1, and $b_{k+1}\le b_{\max}$,

$$\left\|\nabla f(\bar{x}_k)\right\|^2\le 2\left\|\nabla f(\bar{z}_k)\right\|^2+2L^2b_{\max}^2\|\bar{y}_k-\bar{x}_k\|^2.$$

Taking expectation, summing over $k=0,\dots,T-1$, and dividing by T gives

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{x}_k)\right\|^2\le\frac{2}{T}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2+2L^2b_{\max}^2\cdot\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2. \tag{87}$$

Dividing (31) by T gives

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2\le E_1\cdot\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2+E_2\cdot\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left(\tfrac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\right)+E_3\cdot\frac{\sigma^2}{n}, \tag{88}$$

where E_1,E_2,E_3 are the corresponding coefficients in (31). Using

$$\frac{(1-a_{\min})^2}{a_{\min}}\le\frac{1}{a_{\min}},\qquad 1-\alpha_u^2\ge\beta_2(1-\beta_1),$$

together with $1-\alpha_u^2=\beta_2(1-\beta_1)(1+\alpha_u)$ and $(1+\alpha_u^2)/(1+\alpha_u)^2\le 1$, we obtain

$$E_1\le\frac{28\eta^2}{a_{\min}^2(1-\beta_1)^2},\qquad E_2\le\frac{28L^2\eta^2}{a_{\min}^2(1-\beta_1)^2},\qquad E_3\le\frac{6\eta^2}{a_{\min}^2(1-\beta_1)^2}.$$

Now choose $\eta=c/\sqrt{T}$. By Theorem 1,

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{z}_k)\right\|^2\le\frac{C_{z,1}}{\sqrt{T}}+\frac{C_{z,2}}{T}.$$

Also, by Proposition 3 and $C_\Psi=\eta^2 B_\Psi$,

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\Big(\tfrac{1}{n}\left\|Z_k-\bar{Z}_k\right\|_F^2\Big)\le\frac{C_{Z,2}}{T}.$$

Moreover,

$$E_1\le\frac{28c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T},\qquad E_2\le\frac{28L^2c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T},\qquad E_3\le\frac{6c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T}.$$

Substituting these bounds into (88) gives

$$\begin{aligned}\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2\le{}&\frac{28c^2C_{z,1}}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T^{3/2}}+\frac{28c^2C_{z,2}}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T^2}\\&+\frac{28L^2c^2C_{Z,2}}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{1}{T^2}+\frac{6c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{\sigma^2}{n}\cdot\frac{1}{T}.\end{aligned}\tag{89}$$

Since $T^{-2}\le T^{-3/2}$ for $T\ge 1$, we obtain

$$\begin{aligned}\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\bar{y}_k-\bar{x}_k\|^2\le{}&\frac{6c^2}{a_{\min}^2(1-\beta_1)^2}\cdot\frac{\sigma^2}{n}\cdot\frac{1}{T}\\&+\frac{28c^2}{a_{\min}^2(1-\beta_1)^2}\left(C_{z,1}+C_{z,2}+L^2C_{Z,2}\right)\frac{1}{T^{3/2}}.\end{aligned}\tag{90}$$

Finally, substituting (90) into (87) gives

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\left\|\nabla f(\bar{x}_k)\right\|^2\le\frac{C_{x,1}}{\sqrt{T}}+\frac{C_{x,2}}{T}+\frac{C_{x,3}}{T^{3/2}},$$

with $C_{x,1},C_{x,2},C_{x,3}$ defined in (84)–(86). This proves (10). □

C.1 Proof of Corollary 2

Corollary 2 (Linear-speedup bound for the network-average output sequence)**.** *Under the assumptions of Corollary 1, including* (8)*, the choice $\eta=c\sqrt{n/T}$ gives*

$$\frac{1}{T}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{x}_k)\|^2\le\frac{7}{3}\left(\frac{D_1}{\sqrt{nT}}+\frac{D_2 n}{T}\right).\tag{91}$$

Proof. Write $S_{x,T}=T^{-1}\sum_{k=0}^{T-1}\mathbb{E}\|\nabla f(\bar{x}_k)\|^2$, with $S_{z,T}$ defined analogously. Let $E_{Z,T}$ be the left-hand side of (78). Equations (87) and (88), together with the bounds on E_1,E_2,E_3, give

$$S_{x,T}\le(2+q)S_{z,T}+qL^2E_{Z,T}+V,\tag{92}$$

where

$$q=\frac{56L^2b_{\max}^2\tilde{\eta}^2}{a_{\min}^2},\qquad V=\frac{12L^2b_{\max}^2\tilde{\eta}^2\sigma^2}{a_{\min}^2 n}.$$

Write I, N, R for the terms involving the initial objective gap, consensus error, and gradient variance in (6), respectively, so that $S_{z,T} \le I + N + R$. By (8), $q \le (7/24) b_{\max}^2 \beta_2^2 (1-\beta_1)^2 \le 7/24$. Also, $L^2 E_{Z,T} \le N/32$ by (78). Since $L\tilde{\eta} \le 1/8$ and $R \ge 36 L\tilde{\eta}\sigma^2/(a_{\min}^2 n)$, we have $V \le R/24$. Hence

$$\begin{aligned} S_{x,T} &\le \tfrac{55}{24} I + \left(\tfrac{55}{24} + \tfrac{7}{768}\right) N + \tfrac{7}{3} R \\ &\le \tfrac{7}{3}(I + N + R). \end{aligned}$$

For $\eta = c\sqrt{n/T}$, (82) and (83) give $I + R = D_1/\sqrt{nT}$ and $N = D_2 n/T$, which proves (91). □

Remark 13. Keeping the individual terms in (92) gives, for $\eta = c\sqrt{n/T}$,

$$\begin{aligned} S_{x,T} \le \frac{2D_1}{\sqrt{nT}} + \frac{2D_2 n}{T} \\ + \mathcal{O}\left(\frac{\sqrt{n}}{T^{3/2}} + \frac{n^2}{T^2} + \frac{\sigma^2}{T}\right). \end{aligned} \tag{93}$$

The constants are independent of T for fixed problem, network, and algorithm parameters: $q = O(n/T)$, $E_{Z,T} = O(n/T)$, and $V = O(\sigma^2/T)$. Remark 3 gives the threshold at which $D_2 n/T$ is no larger than $D_1/\sqrt{nT}$.

C.2 Squared gradient norm at the local output models

Under the assumptions of Theorem 2,

$$\begin{aligned} \frac{1}{nT}\sum_{k=0}^{T-1}\sum_{i=1}^{n} \mathbb{E}\|\nabla f(x_i^k)\|^2 \le \frac{2}{T}\sum_{k=0}^{T-1} \mathbb{E}\|\nabla f(\bar{x}_k)\|^2 \\ + \frac{2L^2 C_\Psi}{\chi \lambda_X}. \end{aligned} \tag{94}$$

Smoothness of f gives

$$\frac{1}{n}\sum_i \|\nabla f(x_i^k)\|^2 \le 2\|\nabla f(\bar{x}_k)\|^2 + \frac{2L^2}{n}\|X_k - \bar{X}_k\|_F^2.$$

The definition of Ψ_k implies $n^{-1}\|X_k - \bar{X}_k\|_F^2 \le \Psi_k/\lambda_X$. Taking expectations, averaging over k, and using (79) proves (94). Because $C_\Psi = \eta^2 B_\Psi$, the consensus-error term is $O(1/T)$ for $\eta = c/\sqrt{T}$. This bound concerns the global gradient at the local output models and is averaged over agents and iterations; it is not a last-iterate guarantee.

D Additional experiments and hyperparameters

D.1 DPA-1G with one communication round per iteration

DPA uses two communication rounds per iteration, one for the optimization sequence and one for the output sequence. To match the per-iteration communication cost of DSGD, QG-DSGDm, and QG-GUTm, we also study a practical variant called DPA-1G. DPA-1G performs exactly one SENDRECEIVE operation per iteration.

DPA-1G uses an integer period $T_x > 1$. On optimization iterations, agents update the optimization sequence and apply gossip averaging, update the quasi-global momentum buffer, and update the output sequence locally. Every T_x iterations, the method instead performs a gradient-free iteration. It skips the stochastic-gradient computation, keeps the optimization and momentum variables fixed, and uses the single communication round to mix the output sequence.

On optimization iterations, DPA-1G communicates only the locally updated model v_i^k. The output sequence is updated locally as

$$x_i^{k+1} = (1 - a_{k+1}) x_i^k + a_{k+1} y_i^{k+1}.$$

Algorithm 2 DPA-1G with one communication round per iteration

Input: Initial model x_i^0 at agent i, $y_i^0 = x_i^0$, $u_i^0 = 0$, number of iterations T, learning rate η, momentum parameters $\beta_1, \beta_2 \in (0,1)$, schedules $\{a_k\}_{k\geq 1}$ and $\{b_k\}_{k\geq 1}$, mixing weights $\{w_{ij}\}_{j\in\mathcal{N}(i)}$, neighbor set $\mathcal{N}(i)$ including i, and integer period $T_x > 1$.

procedure AGENT-i
  for $k = 0, 1, \ldots, T-1$ **do**
    if $(k+1) \bmod T_x \neq 0$ **then** ▷ Optimization iteration with one communication round
      $z_i^k = b_{k+1} y_i^k + (1 - b_{k+1}) x_i^k$
      Sample $\xi_i^k \sim \mathcal{D}_i$ and compute $g_i^k = \nabla F_i(z_i^k; \xi_i^k)$
      $v_i^k = y_i^k - \eta(\beta_1 u_i^k + g_i^k)$
      SENDRECEIVE(v_i^k)
      $y_i^{k+1} = \sum_{j\in\mathcal{N}(i)} w_{ij} v_j^k$
      $d_i^{k+1} = (y_i^k - y_i^{k+1})/\eta$
      $u_i^{k+1} = (1 - \beta_2) u_i^k + \beta_2 d_i^{k+1}$
      $x_i^{k+1} = (1 - a_{k+1}) x_i^k + a_{k+1} y_i^{k+1}$ ▷ Local averaging
    else ▷ Gradient-free iteration with one communication round
      $y_i^{k+1} = y_i^k$
      $u_i^{k+1} = u_i^k$
      $s_i^k = (1 - a_{k+1}) x_i^k + a_{k+1} y_i^k$
      SENDRECEIVE(s_i^k)
      $x_i^{k+1} = \sum_{j\in\mathcal{N}(i)} w_{ij} s_j^k$
    end if
  end for
  return x_i^T
end procedure

No additional communication is used for x_i^{k+1} on these iterations. On gradient-free iterations, DPA-1G keeps y_i^k and u_i^k fixed, skips the stochastic-gradient computation, and uses the single communication round to compute

$$x_i^{k+1} = \sum_{j\in\mathcal{N}(i)} w_{ij}\big((1 - a_{k+1}) x_j^k + a_{k+1} y_j^k\big).$$

Thus, each agent sends one model-sized vector to each neighbor per iteration. Smaller values of T_x mix the output models more often, while larger values allow more stochastic-gradient updates between gradient-free iterations.

In all reported experiments, we use $T_x = 5$. Thus, four out of every five iterations perform a stochastic-gradient update and apply gossip averaging to the optimization sequence, while the remaining iteration is a gradient-free iteration. The iteration counter continues across epoch boundaries. DPA-1G computes $K - \lfloor K/T_x \rfloor$ mini-batch gradients per agent in the first K iterations. The warmup and coefficient schedules advance on every iteration, including gradient-free iterations.

D.2 Architectures and training details

We follow the architecture and training setup of the public GUT codebase [Aketi et al., 2023]. This includes the model definitions, EvoNorm normalization for the CIFAR models, preprocessing, data augmentation, and the simulated distributed training setup. We use LeNet-5 for Fashion-MNIST, VGG-11 and ResNet-20 for CIFAR-10, and ResNet-20 for CIFAR-100. This matches the setup used for the GUT baselines. LeNet-5 has no normalization layer; VGG-11 uses the benchmark's reduced-width configuration. Torus neighbors are $i \pm 1, i \pm 4$ modulo n. Uniform mixing gives ring, torus, and Dyck self-weights $1/3$, $1/5$ and $1/4$, respectively.

D.3 Hyperparameter tables

For DSGD, QG-DSGDm, and QG-GUTm, we adopt the hyperparameters from the public GUT codebase [Aketi et al., 2023], which we also tested in our experimental settings.

For the equal-epoch non-IID comparisons, Table 2 lists the learning rates for the 16-agent ring. Table 3 gives the learning rates for CIFAR-10/ResNet-20 on the 32-agent ring. Table 4 gives the learning rates for the other 32-agent CIFAR-10 comparisons, using VGG-11 on a ring and ResNet-20 on the torus and Dyck graphs. In these three tables, the last column applies to both DPA and DPA-1G.

We train for 100 epochs on Fashion-MNIST and 200 epochs on CIFAR-10 and CIFAR-100. Each agent uses a mini-batch size of 32. The total mini-batch size across agents is 512 for $n = 16$ and 1024 for $n = 32$. For these comparisons, we use random seeds 0, 1, and 2.

Table 2: Learning rates on the 16-agent ring.

Dataset / Model	α	DSGD	GT	QG-DSGDm	MT	QG-GUTm	DPA / DPA-1G
Fashion-MNIST / LeNet-5	0.1	0.1	0.01	0.1	0.05	0.1	0.1
Fashion-MNIST / LeNet-5	0.01	0.1	0.01	0.1	0.05	0.1	0.1
CIFAR-10 / VGG-11	0.1	0.1	0.1	0.1	0.1	0.1	5.0
CIFAR-10 / VGG-11	0.01	0.1	0.1	0.1	0.05	0.1	5.0
CIFAR-10 / ResNet-20	0.1	0.1	0.1	0.1	0.1	0.1	5.0
CIFAR-10 / ResNet-20	0.01	0.1	0.1	0.1	0.1	0.1	5.0
CIFAR-100 / ResNet-20	0.1	0.1	0.1	0.1	0.1	0.1	5.0
CIFAR-100 / ResNet-20	0.01	0.1	0.1	0.1	0.05	0.1	5.0

Table 3: ResNet-20 learning rates on the 32-agent ring.

Model / Graph	α	DSGD	GT	QG-DSGDm	MT	QG-GUTm	DPA / DPA-1G
ResNet-20 / Ring	0.1	0.1	0.05	0.1	0.05	0.1	5.0
ResNet-20 / Ring	0.01	0.1	0.05	0.1	0.05	0.1	5.0

Table 4: Learning rates for additional 32-agent comparisons.

Model / Graph	α	QG-DSGDm	QG-GUTm	DPA / DPA-1G
VGG-11 / Ring	0.1	0.1	0.1	5.0
VGG-11 / Ring	0.01	0.1	0.1	5.0
ResNet-20 / Torus	0.01	0.1	0.1	5.0
ResNet-20 / Dyck	0.01	0.1	0.1	5.0

For DSGD, GT, QG-DSGDm, MT, and QG-GUTm, we divide the learning rate by 10 after 50% and 75% of training. For DPA and DPA-1G, we increase the learning rate linearly to η during warmup and then keep it constant.

QG-DSGDm and QG-GUTm use a momentum parameter of 0.9. Table 5 lists the QG-GUTm scaling coefficients for the equal-epoch non-IID comparisons.

MT uses a momentum parameter of 0.9. The implementation settings for GT and MT are specified in the released code.

Table 6 lists the training epochs, learning rates, and momentum parameters used in the equal-epoch non-IID comparisons.

For DPA and DPA-1G, we use the exponential coefficient schedules for a_{k+1} and b_{k+1} described in Appendix A.2, with

$$a_{\text{start}} = 0.005, \qquad a_{\text{end}} = 0.0001, \qquad b_{\text{start}} = 0.3, \qquad b_{\text{end}} = 0.1, \qquad q_a = q_b = 1.$$

Let T denote the total number of training iterations. Table 7 gives the coefficient schedule lengths T_a and T_b, and the warmup duration T_w, as fractions of T. For these comparisons, the reference training horizon equals the total training horizon.

Table 5: QG-GUTm scaling coefficients.

Dataset / Model	n	Graph	α	Scaling coefficient
Fashion-MNIST / LeNet-5	16	Ring	0.1	0.01
Fashion-MNIST / LeNet-5	16	Ring	0.01	0.005
CIFAR-10 / VGG-11	16	Ring	0.1	0.08
CIFAR-10 / VGG-11	16	Ring	0.01	0.09
CIFAR-10 / ResNet-20	16	Ring	0.1, 0.01	0.06
CIFAR-100 / ResNet-20	16	Ring	0.1, 0.01	0.005
CIFAR-10 / VGG-11	32	Ring	0.1, 0.01	0.08
CIFAR-10 / ResNet-20	32	Ring	0.1, 0.01	0.04
CIFAR-10 / ResNet-20	32	Torus	0.01	0.05
CIFAR-10 / ResNet-20	32	Dyck	0.01	0.05

Table 6: DPA and DPA-1G hyperparameters.

Dataset / Model	Graph	n	α	Epochs	η	(β_1, β_2)
Fashion-MNIST / LeNet-5	Ring	16	0.1, 0.01	100	0.1	$(0.95, 0.85)$
CIFAR-10 / VGG-11	Ring	16, 32	0.1, 0.01	200	5.0	$(0.95, 0.85)$
CIFAR-10 / ResNet-20	Ring	16, 32	0.1, 0.01	200	5.0	$(0.90, 0.80)$
CIFAR-100 / ResNet-20	Ring	16	0.1, 0.01	200	5.0	$(0.90, 0.80)$
CIFAR-10 / ResNet-20	Torus	32	0.01	200	5.0	$(0.90, 0.70)$
CIFAR-10 / ResNet-20	Dyck	32	0.01	200	5.0	$(0.95, 0.65)$

DPA uses two communication rounds per iteration, one for the optimization sequence and one for the output sequence. For DPA-1G, we set $T_x = 5$. The warmup and coefficient schedules advance on every iteration, including the gradient-free iterations of DPA-1G.

Table 7: Coefficient schedule lengths and warmup.

α	T_a/T	T_b/T	T_w/T
0.1	1.0	0.15	0.25
0.01	1.0	0.30	0.50

The coefficient bounds in Appendix A.1 hold with $a_{\min} = 10^{-4}$, $a_{\max} = 5 \times 10^{-3}$, $b_{\min} = 0.1$, and $b_{\max} = 0.3$.

D.4 Effect of momentum and the b_k schedule

Figure 2 compares DPA with variants without momentum on CIFAR-10/ResNet-20 using a ring topology with $n = 16$ agents and $\alpha = 0.01$. In the legend, labels starting with `DPA_schedule_free` denote variants without momentum. The label with b_k alone uses a decreasing schedule, and the labels with numerical values use fixed coefficients.

DPA has higher test accuracy early in training and at the last epoch than the variants without momentum. Among the variants without momentum, the decreasing b_k schedule gives slightly higher test accuracy than the fixed values tested. Training with $b_k = 0.2$ is unstable in this experiment.

D.5 Extended results

Figure 3 shows the test accuracy curves corresponding to Table 8. Under stronger heterogeneity ($\alpha = 0.01$), DPA-1G has the highest mean LE and L5E, with an L5E of 76.34% versus 76.13% for DPA. At moderate heterogeneity ($\alpha = 0.1$), DPA gives the best mean LE and L5E accuracy, improving over the strongest baseline, QG-GUTm, by 3.43 percentage points in L5E.

Figure 4 and Table 9 report results on the torus and Dyck graphs at $\alpha = 0.01$. DPA achieves the highest mean LE accuracy after choosing the momentum parameters separately for each graph, with

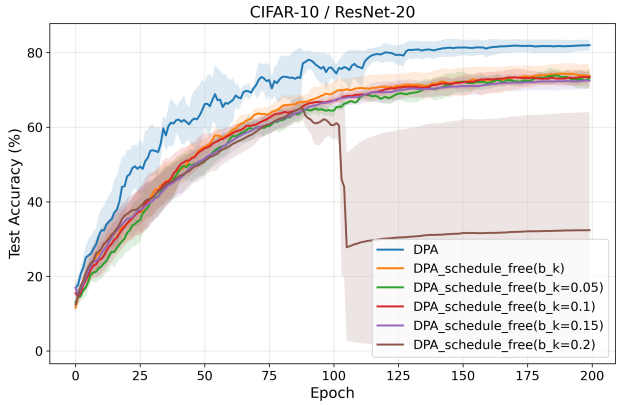


Figure 2: Effect of quasi-global momentum and the b_k schedule on CIFAR-10/ResNet-20 ($n = 16$, ring, $\alpha = 0.01$).

Table 8: CIFAR-10/ResNet-20 test accuracy (%) on a 32-agent ring.

α	Method	LE $\pm$ std	L5E $\pm$ std
0.01	DSGD	21.52 ± 6.74	21.50 ± 6.70
	GT	49.17 ± 0.53	49.16 ± 0.53
	QG-DSGDm	54.52 ± 1.26	55.15 ± 1.34
	MT	44.50 ± 2.66	44.01 ± 3.34
	QG-GUTm	60.35 ± 2.58	60.26 ± 2.97
	DPA-1G	$\mathbf{76.60 \pm 1.81}$	$\mathbf{76.34 \pm 1.81}$
	DPA	76.04 ± 2.35	76.13 ± 2.27
0.1	DSGD	44.85 ± 6.29	45.13 ± 6.31
	GT	52.54 ± 1.03	52.53 ± 1.03
	QG-DSGDm	80.30 ± 2.57	80.32 ± 2.56
	MT	72.66 ± 3.65	72.36 ± 3.50
	QG-GUTm	82.16 ± 1.50	82.14 ± 1.53
	DPA-1G	81.52 ± 3.19	81.48 ± 3.18
	DPA	$\mathbf{85.62 \pm 0.50}$	$\mathbf{85.57 \pm 0.49}$

QG-GUTm close behind. The mean L5E gap between the two methods is 0.46 percentage points on the torus and 0.09 on the Dyck graph, compared with 15.87 on the 32-agent ring.

Figure 5 shows the CIFAR-10/VGG-11 curves from Table 10. We compare the two QG baselines with DPA and DPA-1G at both heterogeneity levels. At $\alpha = 0.01$, DPA gives the best mean LE and L5E accuracy, improving over QG-GUTm by about 8 percentage points in L5E. At $\alpha = 0.1$, QG-GUTm has the highest mean LE and L5E, followed by DPA.

D.6 Comparison at the same communication budget

DPA communicates two model-sized vectors per neighbor per iteration, while DSGD, QG-DSGDm, QG-GUTm, and DPA-1G communicate one. GT and MT communicate approximately two model-sized vectors per neighbor per iteration. We therefore compare 200 epochs of DPA, GT, and MT with separate 400-epoch runs of the methods that communicate one model-sized vector per neighbor per iteration. We apply the baseline learning-rate decay after epochs 100 and 150 in the 200-epoch runs and after epochs 200 and 300 in the 400-epoch runs.

For the 400-epoch DPA-1G runs, we use the same optimizer hyperparameters as in the corresponding 200-epoch runs in Appendix D.3. The reference horizon for the coefficient schedules and warmup remains 200 epochs.

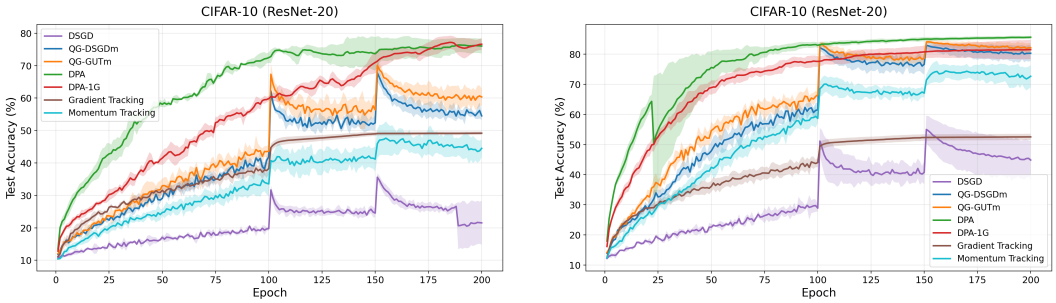


Figure 3: Test accuracy curves on CIFAR-10/ResNet-20 with $n = 32$ agents on the ring topology. Left: $\alpha = 0.01$. Right: $\alpha = 0.1$.

Table 9: CIFAR-10/ResNet-20 test accuracy (%) on torus and Dyck graphs ($n = 32$, $\alpha = 0.01$).

Graph	Method	LE $\pm$ std	L5E $\pm$ std
Torus	QG-DSGDm	84.77 ± 1.05	84.77 ± 1.06
	QG-GUTm	85.92 ± 0.39	85.94 ± 0.36
	DPA-1G	85.35 ± 0.78	85.33 ± 0.79
	DPA	$\mathbf{86.42 \pm 0.51}$	$\mathbf{86.40 \pm 0.53}$
Dyck	QG-DSGDm	71.19 ± 2.50	71.50 ± 2.23
	QG-GUTm	84.77 ± 0.24	84.73 ± 0.24
	DPA-1G	83.54 ± 0.51	83.55 ± 0.51
	DPA	$\mathbf{84.83 \pm 0.48}$	$\mathbf{84.82 \pm 0.50}$

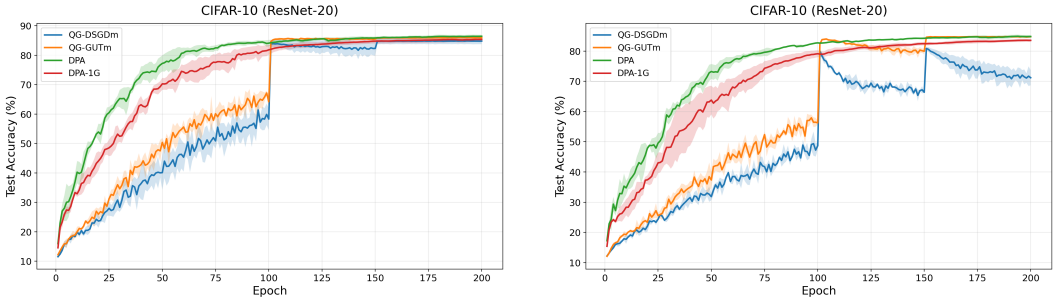


Figure 4: Test accuracy curves on CIFAR-10/ResNet-20 with $n = 32$ agents and $\alpha = 0.01$ on better-connected topologies. Left: Torus. Right: Dyck.

Table 10: CIFAR-10/VGG-11 test accuracy (%) on a 32-agent ring.

α	Method	LE $\pm$ std	L5E $\pm$ std
0.01	QG-DSGDm	57.48 ± 0.43	57.82 ± 1.32
	QG-GUTm	68.53 ± 2.74	69.05 ± 2.47
	DPA-1G	69.26 ± 1.77	69.10 ± 1.84
	DPA	$\mathbf{77.47 \pm 3.99}$	$\mathbf{77.10 \pm 4.24}$
0.1	QG-DSGDm	74.75 ± 1.03	74.38 ± 0.85
	QG-GUTm	$\mathbf{83.54 \pm 0.35}$	$\mathbf{83.56 \pm 0.34}$
	DPA-1G	77.27 ± 2.47	77.21 ± 2.44
	DPA	81.78 ± 0.81	81.75 ± 0.85

We measure communication cost by the cumulative number of tensor bytes sent per agent, summed over all neighbors and excluding received bytes, transport overhead, and post-training averaging. For earlier runs, we estimate the bytes sent by the end of each epoch by multiplying the recorded training total by the fraction of epochs completed. For each seed, we average test accuracy across

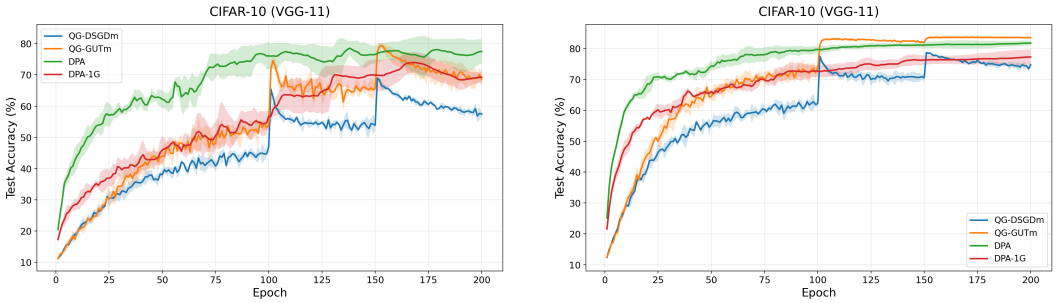
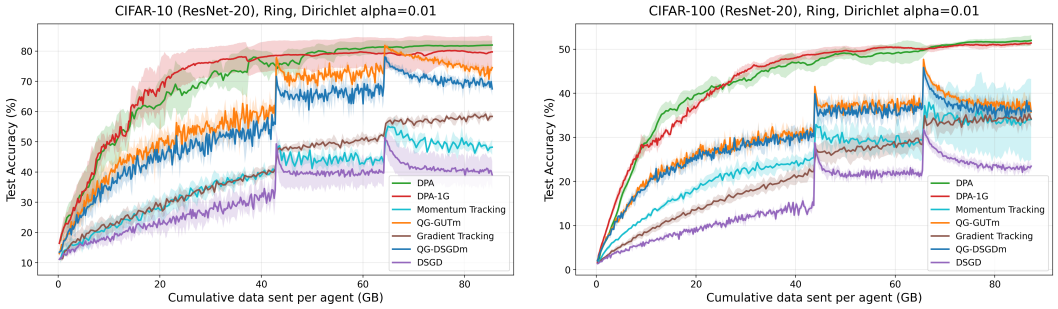


Figure 5: Test accuracy curves on CIFAR-10/VGG-11 with $n = 32$ agents on the ring topology. Left: $\alpha = 0.01$. Right: $\alpha = 0.1$.

agents. We then estimate accuracy at each communication budget using the straight line connecting the two adjacent epoch-end evaluations. We report the mean and standard deviation across three seeds. The full common budgets are $B = 85.45$ GB/agent on CIFAR-10 and $B = 87.28$ GB/agent on CIFAR-100.

Figure 6 shows all seven methods, with mean curves and shaded regions indicating ± 1 standard deviation. Tables 11 and 12 report test accuracy for DPA, DPA-1G, and the baseline with the highest mean at each budget. These entries are accuracies at the specified budgets, not L5E. Boldface marks the highest mean in each column. A run can reach the full common budget between two evaluations, so accuracy at B may differ slightly from last-epoch accuracy (LE).



(a) CIFAR-10 / ResNet-20.

(b) CIFAR-100 / ResNet-20.

Figure 6: Test accuracy versus cumulative data sent per agent (GB) on a 16-agent ring ($\alpha = 0.01$).

Table 11: CIFAR-10/ResNet-20 test accuracy (%) at fractions of $B = 85.45$ GB/agent.

Method	$0.2B$	$0.4B$	$0.6B$	$0.8B$	B
Best baseline (QG-GUTm)	46.61 ± 5.71	56.00 ± 4.58	73.47 ± 2.48	78.39 ± 0.27	74.52 ± 1.10
DPA-1G	$\mathbf{67.22 \pm 7.11}$	$\mathbf{77.40 \pm 4.71}$	78.72 ± 5.07	78.58 ± 6.10	79.73 ± 5.23
DPA	61.99 ± 4.62	73.46 ± 5.69	$\mathbf{80.08 \pm 2.46}$	$\mathbf{81.30 \pm 2.23}$	$\mathbf{82.00 \pm 1.47}$

Table 12: CIFAR-100/ResNet-20 test accuracy (%) at fractions of $B = 87.28$ GB/agent.

Method	$0.2B$	$0.4B$	$0.6B$	$0.8B$	B
Best baseline method	QG-DSGDm	QG-DSGDm	QG-GUTm	QG-GUTm	QG-DSGDm
Best baseline accuracy	23.36 ± 1.54	29.76 ± 0.22	37.82 ± 2.92	39.97 ± 0.31	35.94 ± 1.69
DPA-1G	34.66 ± 0.87	$\mathbf{46.82 \pm 1.28}$	$\mathbf{49.53 \pm 0.97}$	50.51 ± 0.61	51.38 ± 0.37
DPA	$\mathbf{37.99 \pm 2.62}$	45.57 ± 1.14	48.60 ± 1.26	$\mathbf{50.89 \pm 0.40}$	$\mathbf{51.97 \pm 1.18}$

DPA and DPA-1G have the two highest mean accuracies at every reported budget fraction. At the full budget, DPA improves over the best baseline by 7.48 percentage points on CIFAR-10 and 16.03 percentage points on CIFAR-100. The corresponding gains for DPA-1G are 5.21 and 15.44

percentage points, respectively. Our primary comparison uses the full common budget; intermediate columns show test accuracy at smaller budgets under each method's own training schedule. Equal communication does not imply identical learning-rate schedules or equal numbers of gradient evaluations.

D.7 IID evaluation

All four methods train ResNet-20 for 200 epochs on a 16-agent ring, using equal-sized, disjoint random subsets of the CIFAR-10 training data and three random seeds. QG-DSGDm and QG-GUTm use an initial learning rate of 0.1 and a momentum parameter of 0.9. QG-GUTm uses a scaling coefficient of 0.04. DPA and DPA-1G use a base learning rate of 5.0 and momentum parameters $(\beta_1, \beta_2) = (0.90, 0.80)$. For both DPA variants, the coefficient schedules for a_k and b_k follow the $\alpha = 0.1$ settings in Appendix D.3, while learning-rate warmup covers the first 50% of training. With these IID splits, DPA and DPA-1G reach high test accuracy in fewer epochs than the QG baselines under their selected learning-rate schedules (Figure 7) and exceed QG-GUTm by 1.58 and 1.64 percentage points in mean L5E, respectively (Table 13). Table entries report means $\pm$ standard deviations across three seeds. Curves show the means, with shaded regions indicating ± 1 standard deviation.

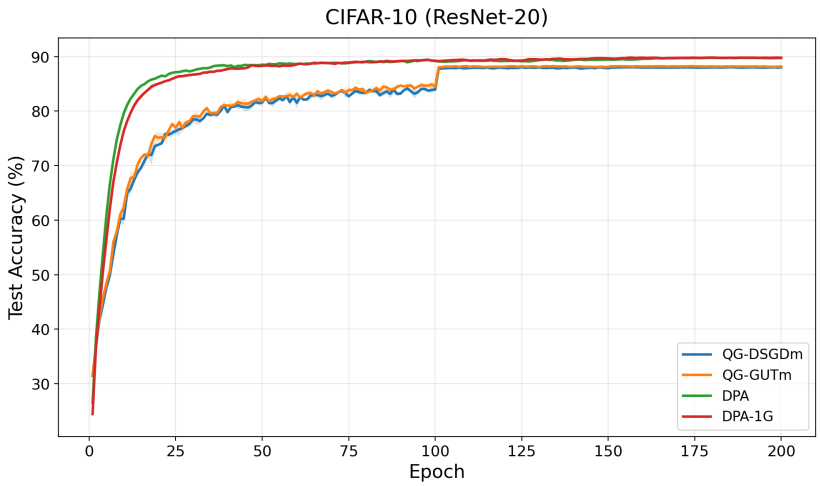


Figure 7: IID CIFAR-10/ResNet-20 test accuracy on a 16-agent ring.

Table 13: IID CIFAR-10/ResNet-20: test accuracy (%) at selected epochs and L5E.

Training progress	QG-DSGDm	QG-GUTm	DPA-1G	DPA
10% (epoch 20)	73.82 ± 0.04	75.10 ± 0.95	85.01 ± 0.32	$\mathbf{86.24 \pm 0.08}$
20% (epoch 40)	79.82 ± 0.47	81.02 ± 0.17	87.71 ± 0.37	$\mathbf{88.25 \pm 0.15}$
40% (epoch 80)	83.36 ± 0.24	83.33 ± 0.76	89.03 ± 0.08	$\mathbf{89.07 \pm 0.31}$
50% (epoch 100)	83.93 ± 0.26	84.47 ± 0.47	$\mathbf{89.22 \pm 0.09}$	89.20 ± 0.29
60% (epoch 120)	87.91 ± 0.16	88.14 ± 0.10	$\mathbf{89.53 \pm 0.19}$	89.21 ± 0.12
80% (epoch 160)	87.97 ± 0.20	88.19 ± 0.13	$\mathbf{89.80 \pm 0.14}$	89.58 ± 0.05
100% (epoch 200)	88.03 ± 0.25	88.17 ± 0.10	$\mathbf{89.78 \pm 0.14}$	89.73 ± 0.17
L5E	88.01 ± 0.26	88.14 ± 0.12	$\mathbf{89.78 \pm 0.14}$	89.72 ± 0.22

D.8 Training time and GPU usage

We run all seven methods sequentially on the same four V100-32GB GPUs. Each run uses ResNet-20 on CIFAR-10, a 16-agent ring, $\alpha = 0.01$, seed 0, and 200 epochs (Table 14). Total time includes training, evaluation, and scheduled checkpoint saving. Training time covers mini-batch training. GPU-hours include all four GPUs. Communication cost and the number of mini-batch gradient evaluations are reported per agent.

Table 14: Training costs on four V100 GPUs (200 epochs, seed 0).

Method	Total time	Training time	GPU-hours	GPU util.	Peak mem.	GB/agent	Grad. evals/agent
DSGD (direct)	45.84 min	34.52 min	3.056	61.83%	4982 MiB	42.72	19,600
QG-DSGDm (direct)	46.81 min	35.51 min	3.121	62.64%	4990 MiB	42.72	19,600
QG-GUTm	115.80 min	104.78 min	7.720	84.40%	5006 MiB	42.72	19,600
GT	51.62 min	40.33 min	3.441	60.98%	4990 MiB	85.45	19,600
MT	52.59 min	41.23 min	3.506	67.42%	4990 MiB	85.45	19,600
DPA-1G	47.17 min	35.79 min	3.145	62.79%	4998 MiB	42.72	15,680
DPA	54.13 min	42.82 min	3.609	67.89%	4996 MiB	85.45	19,600

In this timing experiment, DSGD and QG-DSGDm use direct implementations of their published updates. These implementations avoid the GUT reconstruction and tracking code that still runs in the benchmark when the corresponding coefficients are set to zero. For QG-GUTm, we use the original GUT implementation. We report these single-seed timings separately from the benchmark accuracy results, which use three seeds.

DPA-1G takes 0.8% more total time than direct QG-DSGDm at the same communication cost. DPA takes 15.6% more time than direct QG-DSGDm, but 4.9% and 2.9% more than GT and MT, respectively, which have the same communication cost as DPA. Peak memory varies by less than 0.5% across methods. These measurements apply to the implementations and hardware used in this experiment.

D.9 Effect of post-training averaging

Table 15 compares the five baselines before and after the post-training averaging described in Remark 4. After training, each baseline performs K_{post} communication rounds without taking a gradient step. DPA and DPA-1G do not use this phase; their LE values are shown for reference. Values are means $\pm$ standard deviations across three seeds. Dashes indicate that no post-training averaging was used.

We report this evaluation separately because post-training averaging increases communication cost and changes the local models after training ends. The evaluation after post-training averaging is not part of LE or L5E, and these rounds are excluded from the matched training budgets.

Table 15: Test accuracy (%) before and after post-training averaging ($n = 16$, ring).

	Fashion-MNIST / LeNet-5		CIFAR-10 / ResNet-20		CIFAR-100 / ResNet-20	
Method	LE	After averaging	LE	After averaging	LE	After averaging
			$\alpha = 0.1$			
DSGD	85.605 ± 2.962	88.643 ± 0.906	57.709 ± 3.081	70.041 ± 2.130	25.526 ± 0.475	38.643 ± 0.803
GT	84.517 ± 0.156	84.520 ± 0.156	69.902 ± 2.731	71.925 ± 2.230	41.946 ± 1.353	43.790 ± 0.978
QG-DSGDm	89.717 ± 0.230	89.820 ± 0.193	81.732 ± 1.576	84.430 ± 1.247	44.892 ± 1.846	51.227 ± 3.480
MT	90.064 ± 0.034	90.052 ± 0.025	79.829 ± 1.644	81.453 ± 1.185	49.179 ± 3.432	54.243 ± 1.066
QG-GUTm	89.667 ± 0.002	89.815 ± 0.121	85.338 ± 1.422	86.307 ± 0.645	45.038 ± 0.731	53.368 ± 0.528
DPA-1G	89.33 ± 0.65	—	86.582 ± 0.854	—	53.920 ± 0.546	—
DPA	90.08 ± 0.28	—	86.727 ± 1.103	—	54.120 ± 0.970	—
			$\alpha = 0.01$			
DSGD	73.930 ± 2.985	84.179 ± 0.348	29.786 ± 4.599	48.558 ± 4.961	16.136 ± 0.817	31.525 ± 1.458
GT	84.102 ± 0.226	84.100 ± 0.220	58.381 ± 0.811	61.389 ± 0.741	34.195 ± 1.852	38.968 ± 0.407
QG-DSGDm	76.456 ± 3.624	82.170 ± 2.618	61.470 ± 1.933	74.365 ± 2.636	33.565 ± 0.797	48.484 ± 2.842
MT	82.090 ± 4.793	82.585 ± 4.823	48.204 ± 2.493	43.180 ± 5.789	30.090 ± 2.627	39.278 ± 3.460
QG-GUTm	76.360 ± 5.199	84.459 ± 1.710	66.474 ± 1.744	77.795 ± 1.480	35.544 ± 1.008	48.018 ± 1.883
DPA-1G	85.60 ± 2.46	—	78.570 ± 4.890	—	48.770 ± 1.164	—
DPA	84.73 ± 3.97	—	82.005 ± 1.467	—	51.971 ± 1.180	—

Post-training averaging can improve test accuracy, but the effect varies by method and data distribution. At $\alpha = 0.01$, both DPA and DPA-1G have higher mean accuracy without post-training averaging than all five baselines after post-training averaging on the three tasks shown. On CIFAR-10 and CIFAR-100, DPA exceeds the best baseline after post-training averaging by 4.21 and 3.49 percentage points, respectively. At $\alpha = 0.1$ on CIFAR-100, MT has a slightly higher mean accuracy after

post-training averaging than DPA's mean LE accuracy (54.243% versus 54.120%). These differences in mean accuracy do not establish statistical significance. These evaluations do not show whether the gains persist when local gradient updates resume.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately summarize our methods, theoretical results, and experimental findings.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the main limitations of our method and theoretical analysis in Sections 3 and 5.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: We state the assumptions for all theoretical results and provide complete proofs in the appendix.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the experimental details needed to reproduce our main results in the main text and appendix.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and instructions for reproducing our main experimental results are publicly available. The experiments use publicly available datasets.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide the training and test settings, including data splits and hyperparameters, in the main text and appendix.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the mean and standard deviation across three random seeds for the main accuracy results.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The appendix reports hardware details and resource measurements for all seven methods in one representative setting using a single random seed.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: We have followed the NeurIPS Code of Ethics.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This work focuses on general-purpose decentralized optimization and evaluates the methods on standard image-classification benchmarks.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release datasets or pretrained models.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: We identify the datasets and cite the benchmark code used in this work, but their licenses and terms of use are not fully documented.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released code is documented in the accompanying README, which provides installation instructions and usage examples.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing or research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This work does not involve research with human subjects.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [N/A]

    Justification: LLMs are not used as a component of our methods.