CUMULATIVE-VARIANCE-ADAPTIVE DECENTRALIZED STORM

Anonymous authors
Paper under double-blind review

ABSTRACT

Gradient noise in decentralized optimization may vary along the optimization trajectory, while the smoothness constant is often unknown. We propose Noise-Adaptive Decentralized STORM, which combines locally accumulated direction energy, a two-sample noise proxy, and a neighbor max-envelope to jointly adapt stepsizes and gradient-estimator refresh coefficients. Its horizon-independent fixed hyperparameters require no prior knowledge of the smoothness constant or noise level. Under mean-square smoothness, bounded centered noise, and the stated synchronous-network assumptions, we establish an expected gradient-norm guarantee governed by cumulative expected conditional variance along the optimization trajectory. A uniform variance bound σ^2 yields a rate of $\widetilde{O}(T^{-1/2} + \sqrt{\sigma}\, T^{-1/4})$, improving to $O(T^{-1/2})$ under bounded cumulative variance, including zero noise, with the same hyperparameters. Experiments show robustness to increased stepsize multipliers and fewer queries to prespecified stationarity thresholds than the evaluated baselines in the tested vanishing-noise environments. Classification experiments additionally characterize the accuracy–resource tradeoff.

1 INTRODUCTION

Decentralized learning trains models on distributed data through local computation and neighbor communication, without a central coordinator (Lian et al., 2017). Randomly sampled gradients introduce stochastic noise, while the network must coordinate models and update directions across nodes.

In decentralized stochastic optimization, the smoothness constant is typically unavailable in advance, while gradient noise may vary with the model state. The unknown smoothness constant limits the direct use of stepsize rules that require it. Meanwhile, guarantees that summarize noise only through a uniform worst-case variance bound may fail to distinguish persistently noisy trajectories from those becoming nearly deterministic. This motivates a guarantee that reflects differences in cumulative trajectory noise without reconfiguring the hyperparameters. We therefore ask: *Can decentralized stochastic optimization, without prior knowledge of the smoothness constant and with a fixed hyperparameter configuration, yield convergence guarantees that reflect cumulative noise along the optimization trajectory and improve in regimes with smaller cumulative noise?*

GT-STORM and AdaMDOS improve stochastic optimization efficiency through gradient tracking and variance reduction (Zhang et al., 2021; Huang & Zhao, 2024), while D-NASA reduces reliance on prior knowledge of problem parameters (Li et al., 2025). Our question adds a distinct requirement: the guarantee should reflect cumulative trajectory noise under a fixed configuration. Section 2 discusses the relevant methods and their relationship to this requirement.

Achieving this goal presents two main challenges: coupling between random adaptive scales and estimation errors, and scale mismatch across nodes. First, the scales are constructed from past random observations and are coupled with gradient-estimation errors, so fixed-stepsize error analyses cannot be applied directly. Second, nodes may use different scales, making it difficult to translate error control under local stepsizes into the control required for average-model descent. We address these challenges with *Noise-Adaptive Decentralized STORM*. Our contributions are:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

- **Noise-aware joint adaptation.** We combine local direction energy, a same-point two-sample noise proxy, and a neighbor max-envelope to jointly adapt model stepsizes and estimator refresh coefficients using only neighbor communication. The fixed hyperparameters require no prior knowledge of the smoothness constant or noise level and are independent of the stopping horizon.

- **Cumulative-variance-adaptive convergence.** Under the stated assumptions, we establish a decentralized convergence guarantee governed by cumulative expected conditional variance along the optimization trajectory. We derive uniform scale comparability from the algorithm and connect locally weighted estimation errors to average-model descent. The guarantee yields $\widetilde{O}(T^{-1/2}+\sqrt{\sigma}\,T^{-1/4})$ under uniformly bounded variance and $O(T^{-1/2})$ under bounded cumulative variance, using the same fixed hyperparameters.

- **Empirical benefits and tradeoffs.** Against the evaluated baselines, experiments demonstrate robustness to increased stepsize multipliers and query savings in the tested vanishing-noise environments. Proxy ablation examines the benefit of additional noise observations, while classification tasks characterize accuracy–resource tradeoffs.

2 Related Work

2.1 Decentralized Stochastic Optimization and Gradient Tracking

D-PSGD combines stochastic-gradient updates with neighbor mixing (Lian et al., 2017); GT-STORM adds gradient tracking and recursive momentum to improve stochastic optimization efficiency (Zhang et al., 2021). Other work addresses heterogeneous noise through node-specific sampling (D-NSS and D-NSS-VR; Chen et al., 2026) and heavy-tailed noise through normalization and tracking (GT-NSGDm; Yu et al., 2026). Our analysis instead tracks cumulative conditional variance under bounded centered noise.

2.2 Adaptive Decentralized Optimization

Frameworks for decentralized adaptive methods transfer centralized convergence properties under suitable conditions (Chen et al., 2023). D-NASA combines tracking, moving averages, and normalization to obtain guarantees without prior knowledge of the smoothness constant or network spectral gap (Li et al., 2025). AdaMDOS combines adaptive matrices, momentum-based variance reduction, and tracking for near-optimal stochastic sample complexity under its assumptions (Huang & Zhao, 2024). We distinguish parameter requirements from noise dependence: our target is a guarantee retaining cumulative trajectory variance without using the smoothness constant as an input.

2.3 Noise Adaptivity and Recursive Gradient Estimation

Centralized methods already exploit low-noise conditions: AdaGrad-Norm uses accumulated gradient norms (Ward et al., 2019), while STORM uses recursive momentum without knowing σ (Cutkosky & Orabona, 2019). STORM+ removes the original method's reliance on smoothness and gradient-bound inputs (Levy et al., 2021). META-STORM's same-point, different-sample gradient differences provide a precedent for our noise proxy (Liu et al., 2022). Under conditions including bounded stochastic gradients, OptEMA obtains the averaged-gradient-norm rate $\widetilde{O}(T^{-1/2}+\sqrt{\sigma}\,T^{-1/4})$ without parameterizing effective stepsizes by the smoothness constant (Yuan, 2026). The main technical novelty is not the noise proxy or the σ-dependent rate itself, but a decentralized analysis that derives scale comparability and closes cumulative-variance-dependent descent under heterogeneous, history-dependent node scales.

3 PRELIMINARIES

3.1 PROBLEM FORMULATION AND NOTATION

We consider the nonconvex optimization problem over n nodes:

$$\min_{x\in\mathbb{R}^p} F(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x). \tag{1}$$

Each $f_i : \mathbb{R}^p \to \mathbb{R}$ is finite and differentiable. Node i uses only its local oracle $g_i(x;\xi)$, with fixed sampling law $\xi \sim \mathcal{D}_i$, and neighbor communication. Objectives and sampling laws may differ across nodes; the oracle need not be a sample-loss gradient. Write $[n] = \{1,\dots,n\}$ and $\bar{v}_t := n^{-1}\sum_i v_t^i$ for vector or scalar node averages. We use Euclidean vector norms, spectral matrix norms $\|\cdot\|_2$, and natural logarithms.

3.2 ASSUMPTIONS

Assumption 3.1 (Mean-square smoothness)**.** The oracle is jointly measurable and supports evaluation at two points with the same sample. There exists $L > 0$ such that, for every $i \in [n]$ and $x, y \in \mathbb{R}^p$,

$$\mathbb{E}_{\xi\sim\mathcal{D}_i}\|g_i(x;\xi) - g_i(y;\xi)\|^2 \le L^2\|x-y\|^2. \tag{2}$$

Assumption 3.2 (Fresh unbiased sampling and bounded centered noise)**.** Let $\mathcal{H}_t$ be the σ-algebra generated by the history before the fresh samples in the transition $t \to t+1$, including the query points already determined from that history. Conditionally on $\mathcal{H}_t$, each node draws $\xi_{t+1}^i, \xi_{t+1}^{i\prime}$ independently from $\mathcal{D}_i$, and these pairs are independent across nodes. For every $\mathcal{H}_t$-measurable query point x,

$$\mathbb{E}[g_i(x;\xi_{t+1}^i) \mid \mathcal{H}_t] = \nabla f_i(x). \tag{3}$$

Moreover, there exists $K \ge 0$, uniform over nodes, rounds, and query points, such that

$$\|g_i(x;\xi_{t+1}^i) - \nabla f_i(x)\| \le K \quad \text{a.s.} \tag{4}$$

The same conditions hold for the second fresh sample and at initialization.

Sampling is i.i.d. within each local law, not necessarily identically distributed across nodes. Bounded centered noise is stronger than bounded variance, but permits unbounded true gradients and state-dependent variance. By Jensen's inequality, Assumptions 3.1–3.2 also imply that each f_i and F are L-smooth.

Assumption 3.3 (Objective and initialization)**.** The global objective has a finite lower bound:

$$F_* := \inf_{x\in\mathbb{R}^p} F(x) > -\infty.$$

All nodes start at a common deterministic point x_0, so $x_0^i = x_0$.

Define the initial objective gap and gradient heterogeneity by

$$\Delta_0 := F(x_0) - F_*, \qquad H_0 := \frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(x_0) - \nabla F(x_0)\|^2. \tag{5}$$

Assumption 3.4 (Communication network and synchronous timing)**.** Nodes communicate over a fixed, connected, undirected graph. The mixing matrix $W \in \mathbb{R}^{n\times n}$ satisfies

$$W \ge 0, \qquad W = W^\top, \qquad W\mathbf{1} = \mathbf{1}.$$

Let

$$J := \frac{1}{n}\mathbf{1}\mathbf{1}^\top, \qquad \rho := \|W - J\|_2 < 1,$$

and denote the graph diameter by d_G. The neighbor set including the node itself is $\mathcal{N}_i^+ := \{i\} \cup \{j : W_{ij} > 0\}$.

Scalar scale information propagates synchronously by one hop per round: an envelope sent in round t is available to immediate neighbors for scale updates starting in round $t+1$. The precise propagation rule is given in the algorithm section.

The bounds L, K, ρ, d_G enter the analysis but are not numerical inputs to the updates; W and its neighbor relations specify communication.

3.3 Cumulative Variance and Stationarity

Define the centered stochastic noise and its variance function at node i by

$$\delta_i(x;\xi) := g_i(x;\xi) - \nabla f_i(x), \qquad \varsigma_i^2(x) := \mathbb{E}_{\xi\sim\mathcal{D}_i}\|\delta_i(x;\xi)\|^2. \tag{6}$$

Assumption 3.2 implies $\varsigma_i^2(x) \le K^2$. Since x_{t+1}^i is determined by the history before drawing the fresh sample,

$$\varsigma_i^2(x_{t+1}^i) = \mathbb{E}\left[\|\delta_i(x_{t+1}^i;\xi_{t+1}^i)\|^2 \,\middle|\, \mathcal{H}_t\right]. \tag{7}$$

We therefore define the cumulative expected conditional variance over the first T updates by

$$\mathcal{V}_T := \sum_{t=0}^{T-1}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\varsigma_i^2(x_{t+1}^i)\right]. \tag{8}$$

Unlike the almost-sure noise bound K, $\mathcal{V}_T$ measures accumulated variance along random trajectories and is not an algorithm input. A uniform variance bound $\varsigma_i^2(x) \le \sigma^2$ implies $\mathcal{V}_T \le T\sigma^2$.

We measure stationarity by the expected gradient norm. Let $\widehat{t}$ and I be independent uniform draws from $\{0,\dots,T-1\}$ and $[n]$, respectively, independent of the algorithm's run. Then

$$\mathbb{E}\|\nabla F(x_{\widehat{t}}^I)\| = \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\|\nabla F(x_t^i)\|\right]. \tag{9}$$

Our convergence guarantees concern this first-moment gradient-norm metric, not the squared gradient norm.

4 Algorithm

4.1 Method Overview

Each node maintains a model x_t^i, a recursive gradient estimate u_t^i, and a gradient-tracking direction w_t^i. The local budget B_t^i stores past direction energy and noise proxies; the candidate R_t^i adds the current direction energy. A neighbor max-envelope Q_t^i combines this candidate with propagated information, and $s_t^i = \sqrt{Q_t^i}$ determines the stepsize and estimator refresh coefficient. We define these quantities below, then give the complete procedure in Algorithm 1.

The positive hyperparameters η, s_0, γ remain fixed across noise regimes and are not configured using a prescribed stopping horizon. The actual stepsizes and refresh coefficients depend on s_t^i and adapt to the observations collected by the algorithm.

4.2 Local Cumulative Budget and Noise Proxy

Set $b_0 := s_0^2$, $B_0^i = b_0$, and $a_t^i := \|w_t^i\|^2$. After determining x_{t+1}^i, node i draws two independent fresh samples to construct the same-point proxy and update its local budget:

$$z_{t+1}^i := \frac{1}{2}\left\|g_i(x_{t+1}^i;\xi_{t+1}^i) - g_i(x_{t+1}^i;\xi_{t+1}^{i\prime})\right\|^2, \qquad B_{t+1}^i = B_t^i + a_t^i + \gamma z_{t+1}^i. \tag{10}$$

The difference cancels the true gradient; conditional independence and the factor $1/2$ give $\mathbb{E}[z_{t+1}^i \mid \mathcal{H}_t] = \varsigma_i^2(x_{t+1}^i)$. Thus γ weights an unbiased variance observation against direction energy. Only local observations enter B_t^i, not received envelopes.

4.3 Neighbor Max-Envelope and Adaptive Scale

Initialize $Q_{-1}^i = b_0$. At round t, form $R_t^i := B_t^i + a_t^i$ and incorporate previously received neighbor information:

$$Q_t^i := \max\left\{R_t^i, \max_{j\in\mathcal{N}_i^+} Q_{t-1}^j\right\}, \qquad s_t^i := \sqrt{Q_t^i}. \tag{11}$$

It then sends Q_t^i to its neighbors for use in the next round.

Adding a_t^i makes the current scale respond to the current direction; recording it in B_{t+1}^i retains it for future rounds without double-counting. Self-inclusion in $\mathcal{N}_i^+$ gives $Q_t^i \geq \max\{Q_{t-1}^i, B_t^i + \|w_t^i\|^2\}$, hence nondecreasing scales. Hop-by-hop propagation allows different node scales; the analysis controls their multiplicative mismatch rather than requiring equality.

4.4 Joint Model and Gradient-Estimator Updates

The same scale controls model motion and estimator refresh:

$$\alpha_t^i := \frac{\eta}{s_t^i}, \qquad \beta_t^i := \frac{s_0}{s_t^i}. \tag{12}$$

The relation $\alpha_t^i = (\eta/s_0)\beta_t^i$ connects stepsize-weighted errors to refresh-weighted error budgets. Both coefficients are nonincreasing, with $0 < \beta_t^i \leq 1$ and $\|\alpha_t^i w_t^i\| \leq \eta$; the latter bounds the local optimization step, not the full displacement including mixing. Algorithm 1 specifies the model, estimator, and tracker recursions: neighbor model mixing is followed by the local step, the STORM estimator reuses one sample at two model points, and the tracker mixes the estimator increments. The coefficient β_t^i controls retention of the recursive correction, not an ordinary moving-average weight.

Causal ordering. The current scale s_t^i, stepsize α_t^i, refresh coefficient β_t^i, and query point x_{t+1}^i are determined before drawing the fresh samples and are therefore $\mathcal{H}_t$-measurable. The newly generated z_{t+1}^i does not affect the current stepsize; it influences subsequent scales through B_{t+1}^i. The estimator and proxy share the evaluation $g_i(x_{t+1}^i; \xi_{t+1}^i)$ and need not be independent.

4.5 Complete Procedure and Costs

Algorithm 1 gives the execution order, with T used only as a stopping budget.

Algorithm 1 Noise-Adaptive Decentralized STORM

Input: $T \geq 1$, fixed $\eta, s_0, \gamma > 0$, mixing matrix W, initial point x_0.
Initialize: $b_0 = s_0^2$, $x_0^i = x_0$, $B_0^i = Q_{-1}^i = b_0$ for every i.
Draw independent $\xi_0^i \sim \mathcal{D}_i$; set $u_0^i = g_i(x_0; \xi_0^i)$ and $w_0^i = \sum_j W_{ij} u_0^j$.
for $t = 0, \ldots, T-1$ **do**
  for all $i \in [n]$ **in parallel do**
    Read the locally retained Q_{t-1}^i and receive neighbors' Q_{t-1}^j.
    Use the common value s_0^2 for all these envelopes when $t = 0$.
    $a_t^i = \|w_t^i\|^2, \quad R_t^i = B_t^i + a_t^i$.
    $Q_t^i = \max\{R_t^i, \max_{j \in \mathcal{N}_i^+} Q_{t-1}^j\}, \quad s_t^i = \sqrt{Q_t^i}$.
    $\alpha_t^i = \eta / s_t^i, \quad \beta_t^i = s_0 / s_t^i$.
    Send Q_t^i to neighbors for use in round $t+1$.
    $x_{t+1}^i = \sum_j W_{ij} x_t^j - \alpha_t^i w_t^i$.
    Draw independent $\xi_{t+1}^i, \xi_{t+1}^{i\prime} \sim \mathcal{D}_i$.
    $g_{t+1}^{i,+} = g_i(x_{t+1}^i; \xi_{t+1}^i), \quad g_{t+1}^{i,-} = g_i(x_t^i; \xi_{t+1}^i)$.
    $g_{t+1}^{i,\prime} = g_i(x_{t+1}^i; \xi_{t+1}^{i\prime})$.
    $u_{t+1}^i = g_{t+1}^{i,+} + (1 - \beta_t^i)(u_t^i - g_{t+1}^{i,-})$.
    $z_{t+1}^i = \frac{1}{2}\|g_{t+1}^{i,+} - g_{t+1}^{i,\prime}\|^2$.
    $B_{t+1}^i = B_t^i + a_t^i + \gamma z_{t+1}^i$.
  end for
  Mix $w_{t+1}^i = \sum_j W_{ij}(w_t^j + u_{t+1}^j - u_t^j)$ for every i.
end for
Output: $x_{\widehat{t}}^I$, with independent uniform $\widehat{t} \in \{0, \ldots, T-1\}$ and $I \in [n]$, independent of the run.

Including initialization, T updates use $1 + 3T$ gradient queries, $1 + 2T$ independent sample draws, and $1 + 2T$ neighbor vector mixes per node. The three queries per update use two fresh samples, reusing $g_{t+1}^{i,+}$ for the estimator and proxy. Each node additionally sends one scalar Q_t^i per neighbor

per update, with no global reduction or inner consensus loop. It can accompany the model message but is used for scale updates only in the next round; scalar payload is counted separately.

5 Convergence Analysis

5.1 Cumulative-Variance-Adaptive Convergence

Using $\mathcal{V}_T$, Δ_0, H_0, and the random local output defined in Section 3, we obtain the following guarantee.

Theorem 1 (Cumulative-variance-adaptive convergence)**.** *Under Assumptions 3.1–3.4, run Algorithm 1 with arbitrary fixed $\eta, s_0, \gamma > 0$. For the independent uniform output $x_{\widehat{t}}^I$, every integer $T \geq 1$ satisfies*

$$\mathbb{E}\|\nabla F(x_{\widehat{t}}^I)\| \leq \frac{C_\star}{\sqrt{T}}(1 + \mathcal{V}_T^{1/4})\sqrt{\log(2 + \sqrt{\mathcal{V}_T})}. \tag{13}$$

Here $C_\star < \infty$ depends only on $n, L, K, \rho, d_G, \Delta_0, H_0, \eta, s_0, \gamma$, not on T, a positive noise lower bound, or a variance-growth schedule. An explicit expression and the complete proof appear in Appendix A, particularly (A.56) *and Theorem A.13.*

The result incorporates cumulative trajectory noise into the convergence guarantee. With other problem bounds held fixed, a smaller $\mathcal{V}_T$ yields a better upper bound, without reconfiguring the hyperparameters. Here $\mathcal{V}_T$ averages cumulative conditional variance over random trajectories; it is not the observed proxy sum in a single run.

The theorem permits any fixed positive hyperparameters, but $C_\star$ may depend on their values and need not be uniformly bounded over all such choices. As specified in Section 3, problem constants may enter the guarantee without being inputs to the updates.

5.2 Noise-Regime Rates and Query Complexity

Theorem 1 specializes to three noise regimes.

Corollary 1 (Noise regimes)**.** *Under the conditions of Theorem 1,*

$$\mathbb{E}\|\nabla F(x_{\widehat{t}}^I)\| = \begin{cases} \widetilde{O}(T^{-1/2} + \sqrt{\sigma}\, T^{-1/4}), & \varsigma_i^2(x) \leq \sigma^2 \text{ for all } i, x, \\ O(T^{-1/2}), & \sup_{T \geq 1} \mathcal{V}_T \leq \overline{V} < \infty, \\ O\Big(T^{-1/2+r/4}\sqrt{\log(2+T)}\Big), & \mathcal{V}_T = O(T^r), \quad 0 < r \leq 1. \end{cases}$$

The first condition implies $\mathcal{V}_T \leq T\sigma^2$. The second includes zero noise and finite cumulative expected variance; its constant may depend on $\overline{V}$, but no logarithmic factor grows with T. The third covers sublinear growth for $r < 1$ and at most linear growth for $r = 1$.

Here $\widetilde{O}$ suppresses logarithmic factors. Although Assumption 3.2 implies $\varsigma_i^2(x) \leq K^2$, the theorem retains $\mathcal{V}_T$ rather than replacing it by TK^2; neither σ nor r is needed to configure the algorithm.

Since T updates use $1 + 3T$ gradient queries per node, a sufficient budget for $\mathbb{E}\|\nabla F(x_{\widehat{t}}^I)\| \leq \varepsilon$ is $\widetilde{O}(\varepsilon^{-2} + \sigma^2\varepsilon^{-4})$ under uniformly bounded variance, or $O(\varepsilon^{-2})$ under bounded cumulative variance, with other parameters fixed. These orders apply to both updates and per-node queries; the latter bound's constant may depend on $\overline{V}$. Network-wide queries total $n(1 + 3T)$, and convergence constants retain their node-count and network dependence.

5.3 Proof Overview

The proof addresses three coupled difficulties in order.

1. Derive scale comparability. Envelope monotonicity, normalized local steps, mixing, and bounded centered noise control estimator and tracker increments. Finite-hop propagation then gives $\max_i s_t^i / \min_i s_t^i \leq \kappa$ for a deterministic, time-independent κ (Lemma A.4). This is a derived mismatch bound, not an assumption of equal or nearly equal scales; its proof does not invoke the later descent or budget bounds.

2. Convert local error control into descent control. Self-normalized summation, the STORM recursion, and network mixing bound estimation and network errors in terms of logarithmic budget growth. Scale comparability then converts locally weighted errors to those required by average-model descent (Lemma A.8), retaining the residual due to unequal stepsizes and tracking disagreement rather than treating the average update as a common-stepsize gradient step.

3. Close budget growth through cumulative variance. A pathwise bound connects envelope scales to local budgets without using the ratio bound. Proxy unbiasedness then introduces $\mathcal{V}_T$ into the expected-budget inequalities, which also use the preceding descent and error controls. Solving these inequalities and combining them with weighted descent yields the random-local-output gradient-norm guarantee. Appendix A gives the closure and explicit constants; Appendix D explains the role of nodewise scales.

6 Experiments

We evaluate Algorithm 1 (Ours, also called Neighbor) in four experiment groups.

6.1 Experimental Setup

The synthetic experiments use four-node rings and 20-dimensional nonconvex objectives. We compare Ours with GT-STORM (Zhang et al., 2021), D-NASA (Li et al., 2025), and AdaMDOS (Huang & Zhao, 2024), keeping configurations frozen except for the explicitly varied factor. We report the normalized ergodic global-gradient norm at local iterates,

$$M_T = \frac{1}{Tn} \sum_{t=0}^{T-1} \sum_{i=1}^{n} \frac{\|\nabla F(x_t^i)\|}{\|\nabla F(x_0)\|}. \tag{14}$$

Lower is better. Query costs count actual stochastic-gradient evaluations, including initialization, same-sample evaluations at two points, and proxy queries. Paired-noise and proxy experiments use five problems with twenty sampling seeds per problem; stepsize sensitivity uses five problems with five seeds each. We average seeds within each problem before computing the mean and pointwise descriptive 95% t intervals over five problem means. Table 1 specifies classification and timing statistics; Appendix B gives settings, parameter sources, and accounting.

6.2 Optimization Quality and Query Efficiency under Paired Noise

At objective scale $c = 1$, we compare matched persistent- and vanishing-noise environments. The two oracles have identical conditional variance at the common initialization and identical same-sample gradient increments, hence share a sample-smoothness upper bound. Their principal design difference is the state dependence of the noise. After selecting $\delta = 0.1$ through exploration, we froze the formulas, parameters, budget, and thresholds before confirming the results on independent new problems and sampling seeds. The budget is 48,001 queries per node; see Appendices B.2–B.3.

Under vanishing noise, Ours attains a final mean M_T of 0.000678354, below all three baselines. Under persistent noise, its value 0.016872 is below D-NASA and AdaMDOS but above GT-STORM's 0.012943. Its relative benefit is therefore more pronounced in the tested vanishing-noise setting.

For query efficiency, Ours has the lowest median first-hit query count at all three prespecified stationarity thresholds under vanishing noise (Appendix C.1, Table 5). At $M_T \leq 0.01$, Ours and GT-STORM require medians of 3,262 and 8,569 queries, respectively, a 61.9% reduction. These costs include the additional gradient query needed by the noise proxy. The stochastic phase in the current vanishing-noise construction is short, so the results support finite-budget gains after an early decrease in noise. Duration diagnostics appear in Appendix C.1.

6.3 Parameter Sensitivity and Scale Transfer

At $c = 1$, bounded additive noise level $\sigma = 4$, and 24,001 queries per node, we vary the reference stepsize configuration by $k \in \{1/8, 1/4, 1/2, 1, 2, 4, 8\}$. Ours, D-NASA, and AdaMDOS scale η,

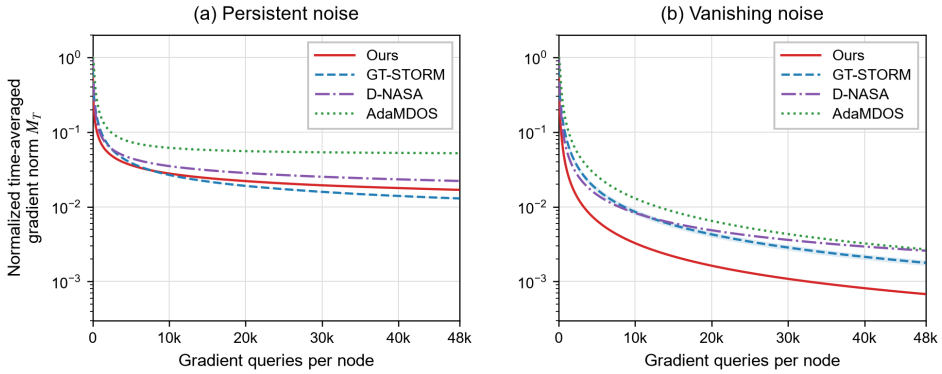


Figure 1: Optimization under matched persistent (left) and vanishing (right) noise. The query axis is linear and the M_T axis logarithmic; means and intervals follow Section 6.1. Curves end at 48,001 queries per node without extrapolation. M_T averages over all preceding iterates, so continued decline need not imply continued instantaneous improvement.

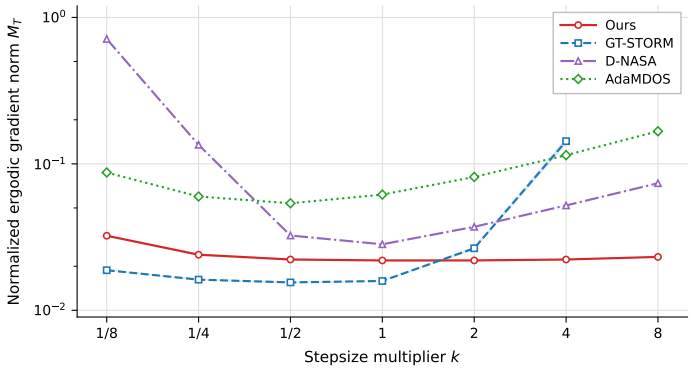


Figure 2: Stepsize-scale sensitivity at a common query budget (coupled GT-STORM). Its $k = 8$ value 5.92×10^{15} exceeds the displayed range and is not plotted. Intervals follow Section 6.1; numerical results are in Appendix C.2.

the stepsize multiplier, and γ_{Ada}, respectively. GT-STORM jointly uses $\eta_0 \leftarrow k\eta_0$ and $\rho_{\mathrm{GT}} \leftarrow \rho_{\mathrm{GT}}/k^2$ to preserve its refresh schedule; a fixed-ρ_{GT} diagnostic is in Appendix C.2.

Figure 2 shows similar optimization quality for Ours at the reference stepsize and its 2, 4, and 8 multiples: the mean at $k = 8$ is only 5.7% above the reference, and it is below all three baselines at the three enlarged multipliers. GT-STORM is better at the reference and smaller multipliers, while Ours deteriorates by about 47.5% at $k = 1/8$. These results mainly show tolerance to increasing the tested stepsize. Frozen-configuration objective-scale transfer, with both normalized and raw gradient measures, appears in Appendix C.3.

6.4 Ablation Study

We compare Full (Ours) and No-proxy in the two paired environments of Section 6.2. No-proxy removes only the noise-proxy term in the local cumulative budget and its independent sample query, retaining the other update rules and parameters without separate tuning. At 48,001 queries per node, Full performs 16,000 updates and No-proxy performs 24,000, so the comparison includes the proxy's additional query cost. This component ablation reuses the same confirmation batch.

In Figure 3, Full reduces the final mean M_T relative to No-proxy by 75.8% under persistent noise and 29.9% under vanishing noise. In the latter setting, the median first-hit cost for $M_T \leq 0.01$ falls

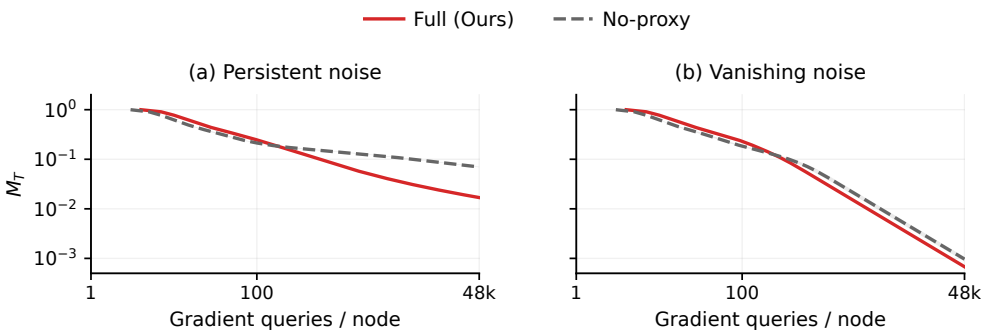


Figure 3: Noise-proxy ablation at equal query budgets: persistent noise (left) and vanishing noise (right). Red solid: Full; gray dashed: No-proxy. Both axes are logarithmic; statistics follow Section 6.1.

from 4,575 to 3,262 queries, a 28.7% reduction. Thus, in this frozen-configuration ablation, the optimization benefit offsets the additional gradient-query cost. Supplementary thresholds and scope are reported in Appendix C.1.

6.5 Classification and Resource Costs

On Fashion-MNIST (Xiao et al., 2017) and KMNIST (Clanuwat et al., 2018), we use four-node rings, a 784–64–10 tanh MLP, Dirichlet-0.1 partitions, and frozen configurations at 6,001 queries per node. We also test D-SGD (Lian et al., 2017) and a stochastic implementation of the GT framework (Qu & Li, 2018) (not GT-STORM). Table 1 reports accuracy, cumulative vector payload, and single-host four-process CPU/Gloo training time.

Table 1: Classification accuracy and resource costs at 6,001 queries per node. Payload is cumulative high-dimensional vector traffic.

Method	Vector payload (GiB/node)	Fashion time (s)	KMNIST time (s)	Fashion accuracy (%)	KMNIST accuracy (%)
D-SGD	4.550	57.22 ± 8.29	42.02 ± 0.56	86.404 ± 0.870	83.384 ± 0.618
GT	9.100	89.12 ± 15.81	83.06 ± 0.89	86.966 ± 0.218	83.676 ± 0.221
D-NASA	13.652	130.63 ± 23.66	114.50 ± 0.92	$\mathbf{87.596 \pm 0.191}$	$\mathbf{83.868 \pm 0.419}$
AdaMDOS	4.551	54.04 ± 8.03	40.61 ± 1.72	86.644 ± 0.191	83.154 ± 0.271
Ours	**3.034**	$\mathbf{42.36 \pm 8.95}$	$\mathbf{32.16 \pm 1.12}$	86.468 ± 0.207	83.198 ± 0.440

Mean $\pm$ sample SD: accuracy over five training seeds; time over five fixed-seed repeats. These reflect different variability. Payload excludes scalars; time includes their processing, but excludes failed attempts and recovery (Appendix B.5).

Ours trails the most accurate method, D-NASA, by 1.128 and 0.670 percentage points, but reduces cumulative vector payload by 33.3%–77.8% versus the four baselines and has the lowest mean controlled training time. Fewer updates and vector mixes at the common query budget are consistent with less neighbor-region time: an accuracy–resource tradeoff, not a per-update communication advantage over every baseline. Details appear in Appendices B.4–B.5 and C.5.

7 Conclusion

Under the stated assumptions, Noise-Adaptive Decentralized STORM adapts to cumulative expected conditional variance using horizon-independent fixed hyperparameters, without knowing the smoothness constant, noise level, or variance-growth schedule. Bounded cumulative variance yields an $O(T^{-1/2})$ expected gradient-norm rate at a random local output. Constants depend on the problem, network, and fixed hyperparameters; universal finite-budget superiority is not guaranteed.

AI USE STATEMENT

Generative AI tools assisted with translation, copy-editing, manuscript organization, experimental code development and debugging, and analysis and visualization of experimental results. They also assisted with checks of experimental protocols and manuscript consistency, and with drafting and checking theoretical arguments. The authors are responsible for the final methods, results, and claims.

REFERENCES

Hongxu Chen, Ke Wei, and Luo Luo. Decentralized non-convex stochastic optimization with heterogeneous variance. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24):20067–20075, 2026. doi: 10.1609/aaai.v40i24.39092. URL https://ojs.aaai.org/index.php/AAAI/article/view/39092.

Xiangyi Chen, Belhal Karimi, Weijie Zhao, and Ping Li. On the convergence of decentralized adaptive gradient methods. In *Proceedings of the 14th Asian Conference on Machine Learning*, volume 189 of *Proceedings of Machine Learning Research*, pp. 217–232. PMLR, 2023. URL https://proceedings.mlr.press/v189/chen23b.html.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical Japanese literature. *arXiv preprint arXiv:1812.01718*, 2018. URL https://arxiv.org/abs/1812.01718.

Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Feihu Huang and Jianyu Zhao. Faster adaptive decentralized learning algorithms. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20490–20525. PMLR, 2024. URL https://proceedings.mlr.press/v235/huang24ah.html.

Kfir Y. Levy, Ali Kavis, and Volkan Cevher. STORM+: Fully adaptive SGD with recursive momentum for nonconvex optimization. In *Advances in Neural Information Processing Systems*, volume 34, pp. 20571–20582, 2021.

Jiaxiang Li, Xuxing Chen, Shiqian Ma, and Mingyi Hong. Tuning-free decentralized nonconvex stochastic optimization. *Pacific Journal of Optimization*, 21(4):643–676, 2025. doi: 10.61208/pjo-2025-009. URL http://www.yokohamapublishers.jp/online-p/PJO/Open/vol21/pjov21n4p643.pdf.

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. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Zijian Liu, Ta Duy Nguyen, Thien Hang Nguyen, Alina Ene, and Huy L. Nguyen. META-STORM: Generalized fully-adaptive variance reduced SGD for unbounded functions. *arXiv preprint arXiv:2209.14853*, 2022. URL https://arxiv.org/abs/2209.14853.

Guannan Qu and Na Li. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2018. doi: 10.1109/TCNS.2017.2698261.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. AdaGrad stepsizes: Sharp convergence over nonconvex landscapes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 6677–6686. PMLR, 2019. URL https://proceedings.mlr.press/v97/ward19a.html.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. URL https://arxiv.org/abs/1708.07747.

Shuhua Yu, Dusan Jakovetic, and Soummya Kar. Decentralized nonconvex optimization under heavy-tailed noise: Normalization and optimal convergence. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2505.03736v2.

Ganzhao Yuan. OptEMA: Adaptive exponential moving average for stochastic optimization with zero-noise optimality. *arXiv preprint arXiv:2603.09923*, 2026. URL https://arxiv.org/abs/2603.09923v4. Version 4.

Xin Zhang, Jia Liu, Zhengyuan Zhu, and Elizabeth S. Bentley. GT-STORM: Taming sample, communication, and memory complexities in decentralized non-convex learning. In *Proceedings of the Twenty-Second International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, pp. 142–151. Association for Computing Machinery, 2021. doi: 10.1145/3466772.3467056.

A Complete Convergence Analysis

We prove Theorem 1 for Algorithm 1 under Assumptions 3.1–3.4. We first collect the notation and recursions used in the proof, then establish auxiliary lemmas and combine them to obtain the convergence bound.

A.1 Notation and Preliminaries

Consider

$$F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x), \qquad x \in \mathbb{R}^p,$$

where each f_i is finite and differentiable. Write $[n] = \{1, \dots, n\}$. For any node-indexed scalar or vector v_t^i, a bar denotes its arithmetic average:

$$\bar{v}_t = \frac{1}{n}\sum_{i=1}^{n} v_t^i. \tag{A.1}$$

This convention applies to scalars as well as to vectors; the pooled raw budget will instead be denoted by $\mathcal{B}_t$. We use the Euclidean norm on vectors and the spectral norm on matrices. Set $J = \mathbf{1}\mathbf{1}^\top/n$. All logarithms are natural.

Let $\mathcal{H}_t$ denote the pre-sampling history in Assumption 3.2, including the states and query points for the transition $t \to t+1$. Write

$$\begin{aligned} \delta_i(x;\xi) &= g_i(x;\xi) - \nabla f_i(x), \\ \varsigma_i^2(x) &= \mathbb{E}\,\|\delta_i(x;\xi)\|^2 \le K^2, \qquad \mathbb{E}_t[\cdot] = \mathbb{E}[\cdot \mid \mathcal{H}_t]. \end{aligned} \tag{A.2}$$

The variance may depend on x and may vanish at some or all nodes.

Throughout, F_* and x_0 are as in Assumption 3.3, and $W, \rho, \mathcal{N}_i^+, d_G$ are as in Assumption 3.4. Jensen's inequality and unbiasedness imply that ∇f_i and ∇F are L-Lipschitz. No bounded true gradients, bounded trajectory-wise gradient heterogeneity, positive noise floor, or node-scale comparison is assumed.

Recursions used in the proof. Fix $\eta, s_0, \gamma > 0$, independently of the stopping horizon, and set $b_0 = s_0^2$. Initialize

$$x_0^i = x_0, \qquad B_0^i = Q_{-1}^i = b_0, \qquad u_0^i = g_i(x_0;\xi_0^i), \qquad w_0^i = \sum_j W_{ij} u_0^j.$$

At iteration $t \ge 0$, define

$$a_t^i = \left\|w_t^i\right\|^2, \qquad R_t^i = B_t^i + a_t^i,$$

$$Q_t^i = \max\left\{R_t^i,\ \max_{j \in \mathcal{N}_i^+} Q_{t-1}^j\right\}, \qquad s_t^i = \sqrt{Q_t^i}, \tag{A.3}$$

$$\alpha_t^i = \frac{\eta}{s_t^i}, \qquad \beta_t^i = \frac{s_0}{s_t^i}. \tag{A.4}$$

In particular, define the scalar averages explicitly as

$$\bar{\alpha}_t = \frac{1}{n}\sum_{i=1}^{n} \alpha_t^i, \qquad \bar{s}_t = \frac{1}{n}\sum_{i=1}^{n} s_t^i. \tag{A.5}$$

In general $\bar{\alpha}_t \ne \eta/\bar{s}_t$; no such equality is used. The model, estimator, and tracker updates are

$$x_{t+1}^i = \sum_j W_{ij} x_t^j - \alpha_t^i w_t^i, \tag{A.6}$$

$$u_{t+1}^i = g_i(x_{t+1}^i;\xi_{t+1}^i) + (1-\beta_t^i)\big[u_t^i - g_i(x_t^i;\xi_{t+1}^i)\big], \tag{A.7}$$

$$w_{t+1}^i = \sum_j W_{ij}\big(w_t^j + u_{t+1}^j - u_t^j\big). \tag{A.8}$$

Using the independent primed sample, set

$$z_{t+1}^i = \frac{1}{2}\left\|g_i(x_{t+1}^i;\xi_{t+1}^i) - g_i(x_{t+1}^i;\xi_{t+1}^{i\prime})\right\|^2, \tag{A.9}$$

$$B_{t+1}^i = B_t^i + a_t^i + \gamma z_{t+1}^i. \tag{A.10}$$

We also write $\bar{z}_{t+1} = n^{-1}\sum_i z_{t+1}^i$. The two unprimed oracle evaluations in (A.7) share a sample. All coefficients and both query points are fixed before that sample pair is drawn. The next envelope may depend on the newly drawn pair.

The raw budget B_t^i accumulates only local contributions; received envelopes are not substituted for, or added to, this budget. Each node retains its previous envelope and sends Q_t^i to its neighbors for use in round $t+1$. The evaluation $g_i(x_{t+1}^i;\xi_{t+1}^i)$ is shared by the estimator and the proxy, giving three oracle evaluations using two fresh samples per update. The accounting is summarized after the proof.

Output and variance budget. For a horizon $T \geq 1$, let $\widehat{t}$ be uniform on $\{0,\dots,T-1\}$ and I uniform on $[n]$, independent of each other and of the run. The output is $x_{\widehat{t}}^I$. Define the cumulative expected conditional variance by

$$\mathcal{V}_T = \sum_{t=0}^{T-1}\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \varsigma_i^2(x_{t+1}^i)\right]. \tag{A.11}$$

The quantity $\mathcal{V}_T$ is deterministic: it averages conditional variances over the random trajectory. It is not a realized sum of squared noise samples and is not the random proxy sum accumulated in a single run. Only the output index is denoted by $\widehat{t}$; R_t^i always denotes the local raw candidate in (A.3).

A.2 Auxiliary Lemmas

We first bound the nodewise scale mismatch, then control estimation and network errors, objective descent, and the cumulative raw budget.

Pathwise bounds and scale comparison. Introduce

$$e_t^i = u_t^i - \nabla f_i(x_t^i), \qquad C_t = \frac{1}{n}\sum_{i=1}^n \left\|x_t^i - \bar{x}_t\right\|^2,$$

$$m_t = \max_i \left\|w_t^i\right\|, \qquad \mathcal{R}_t = \max_i R_t^i. \tag{A.12}$$

All nodewise maxima used in the proof are analytical quantities and need not be computed by the algorithm.

Lemma A.1 (Monotonicity and causal propagation)**.** *For every node and iteration, almost surely,*

$$Q_{t+1}^i \geq Q_t^i, \qquad s_{t+1}^i \geq s_t^i \geq s_0, \qquad 0 < \beta_{t+1}^i \leq \beta_t^i \leq 1, \tag{A.13}$$

$$Q_t^i \geq B_t^i + \left\|w_t^i\right\|^2, \qquad \max_i Q_t^i = \mathcal{R}_t, \tag{A.14}$$

$$\min_i Q_t^i \geq \mathcal{R}_{t-d_G} \qquad (t \geq d_G). \tag{A.15}$$

Proof. Because $i \in \mathcal{N}_i^+$, the maximum defining Q_{t+1}^i contains Q_t^i itself. Hence $Q_{t+1}^i \geq Q_t^i$. Since $Q_0^i \geq b_0 = s_0^2$, monotonicity of the square root and of its reciprocal proves (A.13). This use of local memory does not require $W_{ii} > 0$ and uses no upper bound on the raw budget or the tracking error.

The raw candidates satisfy

$$R_{t+1}^i - R_t^i = a_{t+1}^i + \gamma z_{t+1}^i \geq 0.$$

At $t = 0$, $\max_i Q_0^i = \mathcal{R}_0$, since every $R_0^i \geq b_0$. If $\max_i Q_{t-1}^i = \mathcal{R}_{t-1}$, then every candidate in (A.3) is at most $\mathcal{R}_t$, while each node includes its own current R_t^i. This proves (A.14) by induction.

An envelope value propagates along a graph path by one edge per iteration. For any nodes i, j, there is a path of length at most d_G. Starting with $Q_{t-d_G}^j \geq R_{t-d_G}^j$ and using local retention for any remaining rounds gives $Q_t^i \geq R_{t-d_G}^j$. Maximizing over j proves (A.15). □

Lemma A.2 (Finite-horizon integrability)**.** *For every fixed finite T, the states, scales, and squared error quantities appearing below have finite moments of every order. All conditional expectations and finite sums used in the proof are well defined.*

Proof. By (A.14), $\alpha_t^i \left\|w_t^i\right\| \le \eta$. Nonnegative row-stochastic mixing in (A.6) therefore implies $\max_i \left\|x_t^i - x_0\right\| \le t\eta$. Set

$$G_{\mathrm{init}} = \max_i \left\|\nabla f_i(x_0)\right\|, \qquad \widehat{G}_T = G_{\mathrm{init}} + LT\eta + K.$$

All oracle evaluations used up to iteration T have norm at most $\widehat{G}_T$ almost surely. Recursion in (A.7) gives $\max_i \left\|u_t^i\right\| \le (2t+1)\widehat{G}_T$. Applying convex mixing to (A.8) then gives

$$m_t \le \left[1 + 2t(t+1)\right]\widehat{G}_T \qquad (0 \le t \le T).$$

Moreover $z_{t+1}^i \le 2K^2$. The raw-budget update and (A.14) consequently give deterministic finite-horizon bounds on B_t^i and Q_t^i. Smoothness bounds the objective on the same bounded set of iterates. These bounds establish integrability; they are not substituted into the final convergence rate. □

Define the deterministic constants

$$J_x = \eta\left(1 + \frac{\sqrt{n}(1+\rho)}{1-\rho}\right), \qquad J_u = LJ_x + 4K. \tag{A.16}$$

Lemma A.3 (Model, estimation, and tracking increments)**.** *For every $t \ge 0$ and $i \in [n]$, almost surely,*

$$\sqrt{C_t} \le \frac{\eta}{1-\rho}, \qquad \left\|x_{t+1}^i - x_t^i\right\| \le J_x, \tag{A.17}$$

$$\beta_t^i \left\|e_t^i\right\| \le 2K, \qquad \left\|u_{t+1}^i - u_t^i\right\| \le J_u, \tag{A.18}$$

$$m_{t+1} \le m_t + J_u. \tag{A.19}$$

Proof. Stack the node states as rows of a matrix X_t. The normalized step $\alpha_t^i w_t^i$ has norm at most η at every node. Projection by $I - J$ in (A.6), followed by the Frobenius triangle inequality, gives $\sqrt{C_{t+1}} \le \rho\sqrt{C_t} + \eta$. Since $C_0 = 0$, the first inequality in (A.17) follows. On the disagreement subspace, $\|W - I\|_2 \le 1 + \rho$. Thus the mixing part of a node's displacement has norm at most $(1+\rho)\sqrt{nC_t}$. Adding its normalized step proves the bound by J_x.

Subtracting $\nabla f_i(x_{t+1}^i)$ from (A.7) gives

$$e_{t+1}^i = (1-\beta_t^i)e_t^i + \zeta_{t+1}^i, \qquad \zeta_{t+1}^i = \delta_i(x_{t+1}^i; \xi_{t+1}^i) - (1-\beta_t^i)\delta_i(x_t^i; \xi_{t+1}^i). \tag{A.20}$$

The innovation satisfies $\left\|\zeta_{t+1}^i\right\| \le 2K$ almost surely. Put $\vartheta_t^i = \beta_t^i \left\|e_t^i\right\|$. Initially $\vartheta_0^i \le K$. If $\vartheta_t^i \le 2K$, then

$$\begin{aligned}\vartheta_{t+1}^i &\le \frac{\beta_{t+1}^i}{\beta_t^i}(1-\beta_t^i)\vartheta_t^i + 2K\beta_{t+1}^i \\ &\le 2K\frac{\beta_{t+1}^i}{\beta_t^i}(1-\beta_t^i) + 2K\beta_{t+1}^i = 2K\frac{\beta_{t+1}^i}{\beta_t^i} \le 2K,\end{aligned}$$

where the last inequality follows from Lemma A.1. This is a pathwise induction and does not require β_{t+1}^i to be independent of the current innovation.

Using (A.20),

$$u_{t+1}^i - u_t^i = \nabla f_i(x_{t+1}^i) - \nabla f_i(x_t^i) - \beta_t^i e_t^i + \zeta_{t+1}^i.$$

The preceding bounds and deterministic-gradient smoothness yield J_u. Finally, (A.8) and nonnegative row-stochastic mixing imply

$$\left\|w_{t+1}^i\right\| \le \sum_j W_{ij}\left(\left\|w_t^j\right\| + \left\|u_{t+1}^j - u_t^j\right\|\right) \le m_t + J_u.$$

This bound concerns the full tracking vector, including its disagreement. □

Let $d = d_G$ and define

$$\Lambda_{\mathrm{lag}} = 2d + 3 + \frac{2(d+1)d^2J_u^2 + 2\gamma K^2 d}{b_0},$$
$$\Lambda_{\mathrm{init}} = 1 + \frac{(d+1)(G_{\mathrm{init}} + K + dJ_u)^2 + 2\gamma K^2 d}{b_0},$$
$$\kappa = \sqrt{\max\{\Lambda_{\mathrm{lag}}, \Lambda_{\mathrm{init}}\}}. \tag{A.21}$$

For a single node, take $\kappa = 1$ instead.

Lemma A.4 (Uniform comparison of the actual scales)**.** *Under Assumptions 3.1–3.4, almost surely,*

$$\frac{\max_i s_t^i}{\min_i s_t^i} \le \kappa \qquad \textit{for every } t \ge 0. \tag{A.22}$$

The constant is deterministic and independent of the horizon and any positive noise lower bound.

Proof. The single-node assertion is immediate. Suppose $n > 1$. For $t \ge d$, let $\tau = t - d$. Since $m_\tau^2 \le \mathcal{R}_\tau$, Lemma A.3 gives

$$m_r \le m_\tau + (r-\tau)J_u \le \sqrt{\mathcal{R}_\tau} + dJ_u \qquad (\tau \le r \le t).$$

The exact raw-candidate expansion is

$$R_t^i = R_\tau^i + \sum_{r=\tau+1}^{t} a_r^i + \gamma \sum_{r=\tau+1}^{t} z_r^i.$$

Using $z_r^i \le 2K^2$ and a conservative coefficient $d+1$ for the d direction terms yields

$$\begin{aligned}\mathcal{R}_t &\le \mathcal{R}_\tau + (d+1)\big(\sqrt{\mathcal{R}_\tau} + dJ_u\big)^2 + 2\gamma K^2 d \\ &\le (2d+3)\mathcal{R}_\tau + 2(d+1)d^2J_u^2 + 2\gamma K^2 d.\end{aligned}$$

Since $\min_i Q_t^i \ge \mathcal{R}_\tau \ge b_0$ and $\max_i Q_t^i = \mathcal{R}_t$, their ratio is at most Λ_{lag}.

For $0 \le t < d$, convex mixing at initialization gives $m_0 \le G_{\mathrm{init}} + K$. Thus $m_r \le G_{\mathrm{init}} + K + dJ_u$ for the relevant rounds, and

$$\mathcal{R}_t \le b_0 + (d+1)(G_{\mathrm{init}} + K + dJ_u)^2 + 2\gamma K^2 d.$$

Division by $\min_i Q_t^i \ge b_0$ gives Λ_{init}. Taking square roots proves (A.22). □

For later dependence statements, put

$$\Delta_0 = F(x_0) - F_*, \qquad H_0 = \frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(x_0) - \nabla F(x_0)\|^2. \tag{A.23}$$

Smoothness and the lower bound imply $\|\nabla F(x_0)\|^2 \le 2L\Delta_0$, by evaluating the smoothness inequality at $x_0 - \nabla F(x_0)/L$. Consequently,

$$G_{\mathrm{init}} \le \sqrt{2L\Delta_0} + \sqrt{nH_0}. \tag{A.24}$$

Thus κ depends only on fixed problem, graph, initialization, and algorithm quantities. Lemma A.4 does not use a descent bound or a bound on the cumulative growth of the raw budget.

Self-normalized error and network budgets. Double stochasticity in (A.8) and the initialization give $\bar{w}_t = \bar{u}_t$ for all t. Define

$$\begin{aligned}
\mathcal{B}_t &= \frac{1}{n}\sum_{i=1}^n B_t^i, & A_t &= \frac{1}{n}\sum_{i=1}^n a_t^i,\\
\ell_T &= \log(\mathcal{B}_T/b_0), & h_T &= \mathbb{E}\ell_T,\\
y_T &= \mathbb{E}\sqrt{\mathcal{B}_T}, & a_{\max} &= \eta/s_0,\\
G_t &= \nabla F(\bar{x}_t), & D_t &= \frac{1}{n}\sum_{i=1}^n \left\|w_t^i - \bar{u}_t\right\|^2,\\
S_t &= \frac{1}{n}\sum_{i=1}^n (\alpha_t^i)^2 \left\|w_t^i\right\|^2, & Z_t &= \frac{1}{n}\sum_{i=1}^n \beta_t^i \left\|e_t^i\right\|^2,\\
P_t &= \frac{1}{n}\sum_{i=1}^n (\beta_t^i)^2 \varsigma_i^2(x_{t+1}^i), & \mathcal{M}_{t+1} &= \frac{1}{n}\sum_{i=1}^n \left\|x_{t+1}^i - x_t^i\right\|^2,\\
U_{t+1} &= \frac{1}{n}\sum_{i=1}^n \left\|u_{t+1}^i - u_t^i\right\|^2. & &
\end{aligned} \tag{A.25}$$

The pooled quantities $\mathcal{B}_t, A_t$ and the other averages used in the analysis are not additional communicated statistics.

Introduce constants

$$\begin{aligned}
q &= \frac{1+\rho^2}{2}, & c_W &= \frac{1+\rho^2}{1-\rho^2}, & c_\Delta &= \rho^2 c_W,\\
N_z &= \frac{b_0}{\gamma} + 2K^2, & K_C &= \frac{c_W \eta^2}{1-q}, & K_M &= 2(1+\rho)^2 K_C + 2\eta^2,\\
K_E &= 2N_z + 2L^2 K_M, & K_U &= 26L^2 K_M + 12N_z. & &
\end{aligned} \tag{A.26}$$

For the tracking budget, let

$$d_w = \rho^2(H_0 + K^2), \qquad \mathfrak{D}_0 = \frac{d_w + 4c_\Delta K^2}{1-q}, \qquad \mathfrak{D}_1 = \frac{c_\Delta K_U}{1-q}. \tag{A.27}$$

Lemma A.5 (Direction and proxy budgets)**.** *For every $T \geq 1$,*

$$\sum_{t<T} S_t \leq \eta^2 \ell_T, \qquad \sum_{t<T} \mathbb{E}P_t \leq N_z h_T. \tag{A.28}$$

Proof. For $B > 0$ and $a \geq 0$, $a/(B+a) \leq \log((B+a)/B)$. Hence $Q_t^i \geq B_t^i + a_t^i$ gives

$$(\alpha_t^i)^2 a_t^i \leq \eta^2 \log \frac{B_t^i + a_t^i}{B_t^i}.$$

For $z = z_{t+1}^i$, put $v = \gamma z/(B_t^i + a_t^i)$. Since $0 \leq z \leq 2K^2$, $v \leq 2\gamma K^2/b_0$. The inequality $v \leq (1+v)\log(1+v)$ implies

$$\begin{aligned}
(\beta_t^i)^2 z &\leq \frac{b_0 z}{B_t^i + a_t^i}\\
&\leq \left(\frac{b_0}{\gamma} + 2K^2\right) \log \frac{B_{t+1}^i}{B_t^i + a_t^i}.
\end{aligned}$$

Both logarithmic increments are nonnegative and sum to $\log(B_{t+1}^i/B_t^i)$. Summing over time therefore bounds each of the two budgets by its respective constant times $n^{-1}\sum_i \log(B_T^i/b_0)$, which is at most ℓ_T by Jensen.

The fresh independent pair in (A.9) satisfies

$$\mathbb{E}_t z_{t+1}^i = \varsigma_i^2(x_{t+1}^i). \tag{A.29}$$

Since β_t^i is $\mathcal{H}_t$-measurable, taking expectations of the weighted proxy budget gives the second inequality in (A.28). No independence between the proxy and the updated estimator is needed. □

Lemma A.6 (Consensus and model motion)**.** *Almost surely,*

$$\sum_{t<T} C_t \le K_C \ell_T, \qquad \sum_{t<T} \mathcal{M}_{t+1} \le K_M \ell_T. \tag{A.30}$$

Proof. Projecting (A.6) by $I - J$ and applying Young's inequality gives

$$C_{t+1} \le qC_t + c_W S_t.$$

Indeed, for $\rho > 0$, apply Young with parameter $(1-\rho^2)/(2\rho^2)$ to the mixing and step terms; for $\rho = 0$ the inequality is immediate. Since $C_0 = 0$, summation yields $\sum_{t<T} C_t \le c_W(1-q)^{-1} \sum_{t<T} S_t$. Also

$$\mathcal{M}_{t+1} \le 2(1+\rho)^2 C_t + 2S_t.$$

Apply Lemma A.5 and the definitions in (A.26). The diagonal matrix of local steps is not commuted with the averaging projection in either argument. □

Lemma A.7 (Estimation and tracking budgets)**.** *For every $T \ge 1$,*

$$\sum_{t<T} \mathbb{E} Z_t \le K^2 + K_E h_T, \tag{A.31}$$

$$\sum_{t<T} \mathbb{E} U_{t+1} \le 4K^2 + K_U h_T, \tag{A.32}$$

$$\sum_{t<T} \mathbb{E} D_t \le \mathfrak{D}_0 + \mathfrak{D}_1 h_T. \tag{A.33}$$

Proof. For the two $\mathcal{H}_t$-measurable points x_t^i, x_{t+1}^i, centering the same-sample oracle difference cannot increase its second moment:

$$\mathbb{E}_t \left\| \delta_i(x_{t+1}^i;\xi) - \delta_i(x_t^i;\xi) \right\|^2 \le L^2 \left\| x_{t+1}^i - x_t^i \right\|^2. \tag{A.34}$$

The innovation in (A.20) has conditional mean zero. Its decomposition

$$\zeta_{t+1}^i = \beta_t^i \delta_i(x_{t+1}^i;\xi) + (1-\beta_t^i)\big[\delta_i(x_{t+1}^i;\xi) - \delta_i(x_t^i;\xi)\big]$$

therefore gives

$$\mathbb{E}_t \left\| e_{t+1}^i \right\|^2 \le (1-\beta_t^i) \left\| e_t^i \right\|^2 + 2(\beta_t^i)^2 \varsigma_i^2(x_{t+1}^i) + 2L^2 \left\| x_{t+1}^i - x_t^i \right\|^2. \tag{A.35}$$

Here $(1-\beta_t^i)^2 \le 1 - \beta_t^i$ and $\mathbb{E}_t \langle e_t^i, \zeta_{t+1}^i \rangle = 0$ were used. Average, telescope, and use $\mathbb{E} n^{-1} \sum_i \left\| e_0^i \right\|^2 \le K^2$, followed by Lemmas A.5 and A.6, to obtain (A.31).

Next rewrite the estimator increment as

$$u_{t+1}^i - u_t^i = g_i(x_{t+1}^i;\xi) - g_i(x_t^i;\xi) + \beta_t^i\big[g_i(x_t^i;\xi) - u_t^i\big].$$

Consequently,

$$\mathbb{E}_t \left\| u_{t+1}^i - u_t^i \right\|^2 \le 2L^2 \left\| x_{t+1}^i - x_t^i \right\|^2 + 2(\beta_t^i)^2 \big(\varsigma_i^2(x_t^i) + \left\| e_t^i \right\|^2\big).$$

Equation (A.34) also implies

$$\varsigma_i^2(x_t^i) \le 2\varsigma_i^2(x_{t+1}^i) + 2L^2 \left\| x_{t+1}^i - x_t^i \right\|^2.$$

Using $(\beta_t^i)^2 \le \beta_t^i \le 1$, we obtain

$$\mathbb{E}_t U_{t+1} \le 6L^2 \mathcal{M}_{t+1} + 2Z_t + 4P_t \le 18L^2 \mathcal{M}_{t+1} + 4Z_t + 4P_t. \tag{A.36}$$

Summation proves (A.32) with the conservative constant K_U in (A.26).

For tracking, write the node matrices as W_t and U_t in this paragraph only, so that the fixed mixing matrix remains W. The identity $\bar{w}_t = \bar{u}_t$ gives

$$W_{t+1} - JU_{t+1} = (W - J)\big(W_t - JU_t + U_{t+1} - U_t\big).$$

Young's inequality, as in Lemma A.6, yields

$$\mathbb{E}_t D_{t+1} \le qD_t + c_\Delta \mathbb{E}_t U_{t+1}. \tag{A.37}$$

At initialization, unbiasedness removes the cross term between the deterministic gradient heterogeneity and the centered noise. Projection cannot increase the latter's squared norm, so

$$\mathbb{E} D_0 \le \rho^2(H_0 + K^2) = d_w.$$

Telescope (A.37) and apply (A.32) to obtain (A.33). □

Weighted estimation error and objective descent. Define

$$\chi_t = \left(\frac{1}{n}\sum_{i=1}^{n} s_t^i\right)\left(\frac{1}{n}\sum_{i=1}^{n} 1/s_t^i\right), \qquad \chi_\kappa = \frac{(\kappa+1)^2}{4\kappa}. \tag{A.38}$$

Lemma A.8 (Conversion of local error budgets)**.** *Almost surely,*

$$\bar{\alpha}_t \left\|\bar{e}_t\right\|^2 \le \chi_t a_{\max} Z_t, \qquad 1 \le \chi_t \le \chi_\kappa. \tag{A.39}$$

Consequently,

$$\sum_{t<T} \mathbb{E}\left[\bar{\alpha}_t \left\|\bar{e}_t\right\|^2\right] \le \chi_\kappa a_{\max}(K^2 + K_E h_T). \tag{A.40}$$

Proof. Weighted Cauchy–Schwarz gives

$$\left\|\bar{e}_t\right\|^2 \le \left(\frac{1}{n}\sum_{i=1}^{n} 1/\alpha_t^i\right)\left(\frac{1}{n}\sum_{i=1}^{n} \alpha_t^i \left\|e_t^i\right\|^2\right).$$

Multiply by $\bar{\alpha}_t$ and use $\alpha_t^i = a_{\max}\beta_t^i$ to obtain the first inequality.

Let $m = \min_i s_t^i$ and $M = \max_i s_t^i$. For $s \in [m, M]$, $(s-m)(M-s) \ge 0$ implies $s + mM/s \le m + M$. Writing $a = n^{-1}\sum_i s_t^i$ and $b = n^{-1}\sum_i 1/s_t^i$, we obtain

$$4mMab \le (a + mMb)^2 \le (m+M)^2.$$

Since $M/m \le \kappa$ and $(k+1)^2/(4k)$ is nondecreasing for $k \ge 1$, this proves the upper bound on χ_t. Its lower bound follows from Cauchy–Schwarz. Equation (A.40) now follows from Lemma A.7, without any independence assumption between the scales and the errors. □

Put

$$\Gamma_T = \sum_{t<T} \bar{\alpha}_t \left\|G_t\right\|^2, \qquad \mathcal{U}_T = \sum_{t<T} \bar{\alpha}_t \left\|\bar{u}_t\right\|^2, \tag{A.41}$$

and define

$$\begin{aligned} P_\kappa &= \Delta_0 + a_{\max}\mathfrak{D}_0 + \chi_\kappa a_{\max} K^2, \\ K_\kappa &= a_{\max}(L^2 K_C + \mathfrak{D}_1) + \frac{L\eta^2}{2} + \chi_\kappa a_{\max} K_E. \end{aligned} \tag{A.42}$$

Lemma A.9 (Expected descent budget)**.** *For every $T \ge 1$,*

$$\mathbb{E}\left[\frac{\Gamma_T}{4} + \frac{\mathcal{U}_T}{2}\right] \le P_\kappa + K_\kappa h_T. \tag{A.43}$$

Proof. Set

$$r_t = \frac{1}{n}\sum_{i=1}^{n} \alpha_t^i (w_t^i - \bar{u}_t), \qquad v_t = \bar{x}_t - \bar{x}_{t+1} = \bar{\alpha}_t \bar{u}_t + r_t.$$

Weighted Cauchy–Schwarz and Jensen give

$$\frac{\left\|r_t\right\|^2}{\bar{\alpha}_t} \le a_{\max} D_t, \qquad \left\|v_t\right\|^2 \le S_t. \tag{A.44}$$

Furthermore,

$$\left\|G_t - \bar{u}_t\right\|^2 \le 2\left\|\bar{e}_t\right\|^2 + 2L^2 C_t.$$

Apply smoothness to $F(\bar{x}_t - v_t)$, and use the identity

$$-\langle G_t, \bar{u}_t\rangle = -\tfrac{1}{2}\left\|G_t\right\|^2 - \tfrac{1}{2}\left\|\bar{u}_t\right\|^2 + \tfrac{1}{2}\left\|G_t - \bar{u}_t\right\|^2$$

and Young's inequality

$$-\langle G_t, r_t\rangle \le \frac{\bar{\alpha}_t}{4}\left\|G_t\right\|^2 + \frac{\left\|r_t\right\|^2}{\bar{\alpha}_t}.$$

This gives

$$\frac{\bar{\alpha}_t}{4}\|G_t\|^2 + \frac{\bar{\alpha}_t}{2}\|\bar{u}_t\|^2 \le F(\bar{x}_t) - F(\bar{x}_{t+1}) + \bar{\alpha}_t\|\bar{e}_t\|^2 + \bar{\alpha}_t L^2 C_t + \frac{\|r_t\|^2}{\bar{\alpha}_t} + \frac{L}{2}S_t. \tag{A.45}$$

Sum, take expectations, and use $F \ge F_*$, $\bar{\alpha}_t \le a_{\max}$, and Lemmas A.5, A.6, A.7, and A.8. The resulting constants are exactly (A.42). □

Closure of the cumulative raw budget. Introduce the proxy-only sequence

$$\Pi_t = b_0 + \gamma \sum_{r<t} \bar{z}_{r+1}, \qquad \mathcal{B}_t = \Pi_t + \sum_{r<t} A_r. \tag{A.46}$$

Lemma A.10 (Pathwise raw-budget decomposition)**.** *For every $T \ge 1$, almost surely,*

$$\sqrt{\mathcal{B}_T} \le \sqrt{\Pi_T} + \sum_{t<T} \frac{A_t}{\sqrt{\mathcal{B}_t + A_t}} \tag{A.47}$$

$$\le \sqrt{\Pi_T} + \sqrt{n}\left(\frac{2\mathcal{U}_T}{\eta} + \frac{2}{s_0}\sum_{t<T} D_t\right). \tag{A.48}$$

Proof. First, by Lemma A.1 and nonnegativity,

$$Q_t^i \le \mathcal{R}_t = \max_j (B_t^j + a_t^j) \le \sum_j (B_t^j + a_t^j) = n(\mathcal{B}_t + A_t). \tag{A.49}$$

Taking square roots, reciprocating, multiplying by $a_t^i \ge 0$, and averaging yield

$$\frac{A_t}{\sqrt{\mathcal{B}_t + A_t}} \le \sqrt{n}\frac{1}{n}\sum_{i=1}^n \frac{a_t^i}{s_t^i} = \frac{\sqrt{n}}{\eta}\frac{1}{n}\sum_{i=1}^n \alpha_t^i a_t^i. \tag{A.50}$$

This uses the actual envelope denominator, not an equality between that denominator and the local raw candidate.

For the direction increment,

$$\sqrt{\mathcal{B}_t + A_t} - \sqrt{\mathcal{B}_t} = \frac{A_t}{\sqrt{\mathcal{B}_t + A_t} + \sqrt{\mathcal{B}_t}} \le \frac{A_t}{\sqrt{\mathcal{B}_t + A_t}}. \tag{A.51}$$

For fixed $c \ge 0$, the function $x \mapsto \sqrt{x+c} - \sqrt{x} = c/(\sqrt{x+c} + \sqrt{x})$ is nonincreasing on $(0, \infty)$. Since $\mathcal{B}_t + A_t \ge \Pi_t$, with $c_t = \gamma \bar{z}_{t+1}$ it follows that

$$\sqrt{\mathcal{B}_t + A_t + c_t} - \sqrt{\mathcal{B}_t + A_t} \le \sqrt{\Pi_t + c_t} - \sqrt{\Pi_t} = \sqrt{\Pi_{t+1}} - \sqrt{\Pi_t}. \tag{A.52}$$

Add (A.51) and (A.52) and telescope, using $\mathcal{B}_0 = \Pi_0 = b_0$, to obtain (A.47).

Finally,

$$\frac{1}{n}\sum_{i=1}^n \alpha_t^i a_t^i \le 2\bar{\alpha}_t\|\bar{u}_t\|^2 + 2a_{\max} D_t.$$

Combine this with (A.50) and (A.47) to obtain (A.48). All steps are pathwise; this lemma does not require the scale-ratio bound. □

Define

$$N_T = \sqrt{b_0 + \gamma \mathcal{V}_T},$$
$$d_0 = \sqrt{n}\left(\frac{4P_\kappa}{\eta} + \frac{2\mathfrak{D}_0}{s_0}\right),$$
$$d_1 = \sqrt{n}\left(\frac{4K_\kappa}{\eta} + \frac{2\mathfrak{D}_1}{s_0}\right). \tag{A.53}$$

Lemma A.11 (Expected closure and explicit envelopes)**.** *For every $T \geq 1$,*

$$y_T \leq N_T + d_0 + d_1 h_T, \qquad h_T \leq 2\log(y_T/s_0). \tag{A.54}$$

In particular, set

$$\begin{aligned} A_T^{\mathrm{env}} &= \frac{N_T + d_0}{s_0}, & b &= \frac{2d_1}{s_0}, \\ V_T^{\mathrm{env}} &= 2A_T^{\mathrm{env}} + 2b\log(1+2b), & Y_T &= s_0 V_T^{\mathrm{env}}, \\ h_T^{\mathrm{env}} &= 2\log V_T^{\mathrm{env}}. \end{aligned} \tag{A.55}$$

Then $y_T \leq Y_T$ and $h_T \leq h_T^{\mathrm{env}}$.

Proof. The proxy compensator (A.29) gives $\mathbb{E}\Pi_T = b_0 + \gamma \mathcal{V}_T$, and Jensen yields $\mathbb{E}\sqrt{\Pi_T} \leq N_T$. Taking expectations in (A.48), and using (A.33) and (A.43), gives

$$\begin{aligned} y_T &\leq N_T + \sqrt{n}\left(\frac{2\mathbb{E}\mathcal{U}_T}{\eta} + \frac{2}{s_0}\sum_{t<T}\mathbb{E}D_t\right) \\ &\leq N_T + \sqrt{n}\left(\frac{4(P_\kappa + K_\kappa h_T)}{\eta} + \frac{2(\mathfrak{D}_0 + \mathfrak{D}_1 h_T)}{s_0}\right) \\ &= N_T + d_0 + d_1 h_T. \end{aligned}$$

Here $\Gamma_T \geq 0$ was used to extract a bound on $\mathbb{E}\mathcal{U}_T$ from (A.43). There is no factorization of correlated random quantities. Also

$$h_T = 2\mathbb{E}\log\left(\frac{\sqrt{\mathcal{B}_T}}{s_0}\right) \leq 2\log\left(\frac{y_T}{s_0}\right).$$

Put $Y = y_T/s_0 \geq 1$. The preceding inequalities imply $Y \leq A_T^{\mathrm{env}} + b\log Y$. For $b > 0$, concavity of the logarithm gives

$$\log Y \leq \frac{Y}{2b} + \log(2b) - 1.$$

Substitution and rearrangement yield

$$Y \leq 2A_T^{\mathrm{env}} + 2b\big(\log(2b) - 1\big) \leq 2A_T^{\mathrm{env}} + 2b\log(1+2b) = V_T^{\mathrm{env}}.$$

The case $b = 0$ is immediate. This proves the explicit bounds. □

The pathwise decomposition in Lemma A.10 does not use κ. The expected closure in Lemma A.11 does use κ through the descent constants P_κ, K_κ. There is no reverse dependence: Lemma A.4 was proved using only local normalization, monotone refresh, bounded noise, and finite-hop propagation, without a cumulative-budget or expected descent bound.

A.3 Proof of the Main Convergence Theorem

Using the constants d_0, d_1 in (A.53) and $b = 2d_1/s_0$ in (A.55), define

$$\begin{aligned} c_Y &= 2s_0 + 2d_0 + 2s_0 b\log(1+2b) + 2\sqrt{\gamma}, \\ c_h &= 2 + \frac{2\log(c_Y/s_0)}{\log 2}, \\ c_P &= \frac{P_\kappa}{\log 2} + K_\kappa c_h, \\ C_\star &= 2\sqrt{\frac{\sqrt{n}\, c_Y c_P}{\eta}} + L\sqrt{K_C c_h}. \end{aligned} \tag{A.56}$$

All four constants are finite, nonnegative, and independent of T. In particular, $c_Y \geq 2s_0$, so the logarithm in c_h is well defined and $c_h > 0$. These constants need not be small.

Lemma A.12 (Explicit time-uniform envelope bounds)**.** *Set* $\mathcal{L}_T = \log(2+\sqrt{\mathcal{V}_T})$*. For every* $T \geq 1$*,*

$$Y_T \leq c_Y(1+\sqrt{\mathcal{V}_T}), \tag{A.57}$$

$$h_T^{\mathrm{env}} \leq c_h \mathcal{L}_T, \tag{A.58}$$

$$P_\kappa + K_\kappa h_T^{\mathrm{env}} \leq c_P \mathcal{L}_T. \tag{A.59}$$

Proof. Write $v = \sqrt{\mathcal{V}_T} \geq 0$. Since $N_T = \sqrt{s_0^2 + \gamma v^2} \leq s_0 + \sqrt{\gamma}\, v$, the explicit definition of Y_T gives

$$\begin{aligned} Y_T &= 2(N_T + d_0) + 2s_0 b \log(1+2b) \\ &\leq 2s_0 + 2d_0 + 2s_0 b \log(1+2b) + 2\sqrt{\gamma}\, v \\ &\leq c_Y(1+v). \end{aligned}$$

This proves (A.57). Next, using $h_T^{\mathrm{env}} = 2\log(Y_T/s_0)$ and $\mathcal{L}_T \geq \log 2$,

$$\begin{aligned} h_T^{\mathrm{env}} &\leq 2\log(c_Y/s_0) + 2\log(1+v) \\ &\leq \left(\frac{2\log(c_Y/s_0)}{\log 2} + 2\right)\log(2+v) = c_h \mathcal{L}_T. \end{aligned}$$

Finally, $P_\kappa, K_\kappa \geq 0$ and $1 \leq \mathcal{L}_T/\log 2$ imply

$$P_\kappa + K_\kappa h_T^{\mathrm{env}} \leq \left(\frac{P_\kappa}{\log 2} + K_\kappa c_h\right)\mathcal{L}_T = c_P \mathcal{L}_T.$$

Every inequality holds for all $\mathcal{V}_T \geq 0$, including zero; no asymptotic threshold is required. □

Theorem A.13 (Explicit form of Theorem 1)**.** *Suppose Assumptions 3.1–3.4 hold. For the iteration* (A.3)–(A.10)*, with arbitrary fixed* $\eta, s_0, \gamma > 0$*, and the independent uniform output* $x_{\hat{t}}^I$*, every integer* $T \geq 1$ *satisfies*

$$\mathbb{E}\left\|\nabla F(x_{\hat{t}}^I)\right\| \leq 2\sqrt{\frac{\sqrt{n}\, Y_T(P_\kappa + K_\kappa h_T^{\mathrm{env}})}{\eta T}} + L\sqrt{\frac{K_C h_T^{\mathrm{env}}}{T}}. \tag{A.60}$$

In particular, with the explicit constant $C_\star$ *in* (A.56)*,*

$$\mathbb{E}\left\|\nabla F(x_{\hat{t}}^I)\right\| \leq \frac{C_\star}{\sqrt{T}}(1+\mathcal{V}_T^{1/4})\sqrt{\log(2+\sqrt{\mathcal{V}_T})}. \tag{A.61}$$

The constant may depend on $n, L, K, \rho, d_G, \Delta_0, H_0, \eta, s_0, \gamma$*, but not on a positive noise lower bound or a variance-growth schedule. The problem and graph are held fixed as* T *varies.*

Proof. For $t < T$, Lemma A.1 and the nonnegative raw-budget increments imply

$$Q_t^i \leq \max_j(B_t^j + a_t^j) \leq \sum_j B_T^j = n\mathcal{B}_T.$$

Consequently,

$$\frac{1}{\bar{\alpha}_t} \leq \frac{\sqrt{n}\sqrt{\mathcal{B}_T}}{\eta}. \tag{A.62}$$

Cauchy–Schwarz over the time index gives, pathwise,

$$\begin{aligned} \frac{1}{T}\sum_{t<T}\|G_t\| &\leq \frac{1}{T}\left(\sum_{t<T}\bar{\alpha}_t\|G_t\|^2 \sum_{t<T}\frac{1}{\bar{\alpha}_t}\right)^{1/2} \\ &\leq \left(\frac{\sqrt{n}\sqrt{\mathcal{B}_T}\,\Gamma_T}{\eta T}\right)^{1/2}. \end{aligned}$$

Now apply Cauchy–Schwarz in probability:

$$\mathbb{E}\sqrt{\sqrt{\mathcal{B}_T}\,\Gamma_T} \leq \sqrt{\mathbb{E}\sqrt{\mathcal{B}_T}\,\mathbb{E}\Gamma_T} \leq 2\sqrt{y_T(P_\kappa + K_\kappa h_T)}.$$

No independence between $\mathcal{B}_T$ and Γ_T is required. Therefore

$$\frac{1}{T}\sum_{t<T}\mathbb{E}\,\|G_t\| \le 2\sqrt{\frac{\sqrt{n}\,y_T(P_\kappa + K_\kappa h_T)}{\eta T}}. \tag{A.63}$$

Smoothness also yields

$$\frac{1}{n}\sum_{i=1}^{n}\left\|\nabla F(x_t^i)\right\| \le \|G_t\| + L\sqrt{C_t}.$$

Average over time and apply Cauchy–Schwarz, Jensen, and (A.30) to obtain

$$\frac{L}{T}\sum_{t<T}\mathbb{E}\sqrt{C_t} \le L\sqrt{\frac{1}{T}\sum_{t<T}\mathbb{E}C_t} \le L\sqrt{\frac{K_C h_T}{T}}.$$

Combining these inequalities with (A.63) bounds $T^{-1}\sum_{t<T}\mathbb{E}[n^{-1}\sum_i \left\|\nabla F(x_t^i)\right\|]$. By the independence and uniformity of $\widehat{t}, I$, this quantity equals $\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\|$. Substituting $y_T \le Y_T$ and $h_T \le h_T^{\mathrm{env}}$ proves (A.60).

For the explicit rate, apply Lemma A.12 to (A.60) to obtain

$$\begin{aligned}\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\| &\le \left[2\sqrt{\frac{\sqrt{n}\,c_Y c_P}{\eta}}\sqrt{1+\sqrt{\mathcal{V}_T}} + L\sqrt{K_C c_h}\right]\sqrt{\frac{\mathcal{L}_T}{T}} \\ &\le C_\star(1+\mathcal{V}_T^{1/4})\sqrt{\frac{\mathcal{L}_T}{T}}.\end{aligned}$$

The last step uses $\sqrt{1+\sqrt{V}} \le 1 + V^{1/4}$ and $1 \le 1 + V^{1/4}$ for $V \ge 0$. This proves (A.61) for every $T \ge 1$ with the stated constant. Equation (A.24) gives the stated dependence of $C_\star$. □

A.4 Noise Regimes and Interpretation

The preceding theorem gives the following consequences for different growth rates of the cumulative expected conditional variance.

Corollary A.14 (Noise regimes)**.** *Under the hypotheses of Theorem A.13:*

1. *If* $\sup_T \mathcal{V}_T < \infty$*, then* $\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\| = O(T^{-1/2})$*, without a growing logarithmic factor in* T*. This includes exact noiselessness.*
2. *If* $\varsigma_i^2(x) \le \sigma^2$ *uniformly, then*
$$\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\| \le C_\star\big(T^{-1/2} + \sqrt{\sigma}\,T^{-1/4}\big)\sqrt{\log(2+\sigma\sqrt{T})}.$$
3. *If* $\mathcal{V}_T = O(T^r)$ *for* $0 < r \le 1$*, then*
$$\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\| = O\big(T^{-1/2+r/4}\sqrt{\log(2+T)}\big).$$

Proof. For the first claim, substitute the uniform bound on $\mathcal{V}_T$ into (A.61). For the second, use $\mathcal{V}_T \le \sigma^2 T$ and monotonicity of the right side of (A.61) in $\mathcal{V}_T$. The third follows by the same substitution with $\mathcal{V}_T = O(T^r)$. □

Noise adaptation and assumptions. The update requires neither L nor a noise magnitude or noise-evolution schedule as an input. Under the stated assumptions, its stationarity guarantee improves as the cumulative expected conditional variance $\mathcal{V}_T$ decreases, with all other bounds fixed. This is the sense of noise adaptation established by the theorem. The fixed positive algorithm parameters η, s_0, γ remain user-chosen; the theorem does not assert optimal performance without hyperparameter tuning.

The prefactor $C_\star$ still depends on the almost-sure centered-noise bound K, the smoothness bound, the network, and initialization, through the displayed constants. Thus the result is not a complexity bound that depends only on realized variance, nor one independent of worst-case noise bounds. A comparison across different noise laws must either keep common upper bounds for these quantities or explicitly track their changes. Likewise, not requiring L as an algorithm input does not mean that smoothness is absent from the assumptions or the rate constant. The proof uses almost-sure bounded centered noise, same-sample mean-square smoothness, an independent primed sample, and a fixed synchronous symmetric doubly stochastic network. These are assumptions, not quantities that must be numerically supplied to the update. The fixed sampling laws can yield state-dependent variances along the iterates; arbitrary externally time-varying sampling laws are not covered by the stated sampling assumption.

Stationarity and cost units. Theorem A.13 bounds the first moment $\mathbb{E}\left\|\nabla F(x_{\widehat{t}}^I)\right\|$ in terms of parameter updates T. Any comparison to another result must match the stationarity metric and output rule, the assumptions, and the computational and communication units. Here T complete updates correspond to $1 + 3T$ oracle evaluations and $1 + 2T$ fresh samples per node; a same-sample double evaluation is not two independent samples. The additional primed evaluation and the neighbor scalar messages must not be omitted from the accounting. Including initialization, T updates also use $1+2T$ vector mixes per node, with one scalar sent to each neighbor at each update. Neither L, K, σ, d_G nor a scale-ratio bound is an algorithm input.

In particular, a bound on $\mathbb{E}\,\|\nabla F(x)\|^2$ implies a first-moment bound by Jensen's inequality, but a bound on $\mathbb{E}\,\|\nabla F(x)\|$ alone does not imply the corresponding squared-gradient bound. Comparing only the exponents displayed in the two types of bounds is therefore invalid. No dominance over another algorithm's rate constants or empirical performance is asserted here.

Scope of the guarantee. The conclusion concerns the first moment of the global gradient norm at a uniformly sampled local iterate. It is not a squared-gradient or last-iterate rate. Lemma A.4 establishes a uniformly bounded multiplicative mismatch between node scales, not that their ratios are close to one, converge to one, or are equal at any given time. The ratio constant need not be small, and the rate constant need not be uniform in graph size or diameter. The logarithmic factor in (A.61) is retained. Neighbor scalar communication is part of the algorithm; the result does not cover a variant with no scalar exchange, arbitrary asynchronous delays, or a replacement of bounded centered noise by a finite-variance assumption alone.

B Experimental Settings and Reproducibility Details

This appendix collects the settings, parameter sources, and statistical protocols for the main and supplementary experiments. Supplementary results are in Appendix C.

B.1 Evaluation Protocol and Parameter Sources

The synthetic experiments use the normalized ergodic global-gradient norm at local iterates in Equation (14). Each trajectory is normalized by its problem's initial global-gradient norm before aggregation across instances. Here T is each method's own update count: a common query budget does not imply equal update counts, sample counts, training times, or optimization quality. Final instantaneous gradients and tail averages are auxiliary diagnostics rather than replacements for the full-trajectory M_T.

Table 2: Gradient-query accounting for the synthetic experiments.

Method	Queries/node for T updates	Notes
Ours	$1 + 3T$	Three evaluations; two fresh samples
No-proxy	$1 + 2T$	Same-sample two-point evaluations
GT-STORM	$1 + 2T$	Distinct from classification GT
D-NASA	T	One per update; no initialization query
AdaMDOS	$1 + 2T$	Initialization and two-point evaluations

The current Neighbor implementation was validated in the reference environment $(c, \sigma) = (1, 4)$. Screening evaluated 64 candidates on two validation problems at 3,001 queries each; the top eight were evaluated on three other validation problems at 24,001 queries each. The candidate pool was inherited from the former Global search, and current-version validation selected the same effective parameters; they were not copied without validation. External baselines retain their historical reference selections, whose complete tuning histories are not asserted to be identical.

Table 3: Reference configurations for synthetic problems.

Method	Frozen configuration
Ours	$\eta = 4.655282388394825,\ s_0 = 0.326484740386302$ $\gamma = 0.3923487377042116$
GT-STORM	$\eta_0 = 0.9400799554943875,\ \rho_{\mathrm{GT}} = 0.3524732954287214$
D-NASA	Stepsize multiplier 0.8592645446505197
AdaMDOS	$\gamma_{\mathrm{Ada}} = 0.03,\ \rho_{\mathrm{Ada}} = 0.01$; relaxation, momentum, and second-moment decay all 0.9
No-proxy	Inherit Ours η, s_0; set proxy coefficient $\gamma = 0$

Problem/seed splits and repetition counts are given in Appendix B.3; classification and timing protocols are in Appendices B.4 and B.5.

B.2 Synthetic Objectives and Noise

Sections 6.2–6.4 use four-node rings and the following nonconvex objective family in dimension $p = 20$:

$$\phi(x) = \sum_{k=1}^{p} \big[0.1x_k^2 + \omega_k(1 - \cos x_k)\big], \qquad f_i^{(c)}(x) = c\big[\phi(x) + v_i^\top x\big]. \tag{B.1}$$

For each problem, the ω_k are independent uniform draws on $[0.35, 0.55]$. We center independent standard Gaussian node vectors and rescale them jointly so that $\sum_i v_i = 0$ and $\max_i \|v_i\| = 0.3$. Thus $F^{(c)}(x) = c\phi(x)$. We use $x_0 = 2.2\mathbf{1}$ and $c = 1$ unless performing objective-scale transfer, where c multiplies all deterministic terms while the additive noise level stays fixed. Below, f_i denotes the local objective at the chosen c. The updates do not receive the test environment's smoothness constant, noise bound, or future noise evolution.

Bounded additive noise is used for objective-scale transfer, stepsize sensitivity, and the constant-noise scan. Each coordinate is an independent Rademacher sign multiplied by $\sigma/\sqrt{p}$. Setting $\sigma = 0$ removes this perturbation and returns the true local gradient. Two-point evaluations reuse one sample; the additional proxy sample is independent. This all-time zero-noise setting differs from the state-dependent vanishing noise below.

Let $a = 2.2$ and $\delta = 0.1$, and define componentwise

$$h_\delta(x)_k = \frac{a^2 + \delta^2}{2a^2} \frac{x_k^2}{x_k^2 + \delta^2}. \tag{B.2}$$

For a zero-mean Rademacher vector r, the two oracles are

$$g_i^{\mathrm{van}}(x; r) = \nabla f_i(x) + \frac{4}{\sqrt{p}} h_\delta(x) \odot r, \tag{B.3}$$

$$g_i^{\mathrm{per}}(x; r) = \nabla f_i(x) + \frac{4}{\sqrt{p}} (h_\delta(x) - \mathbf{1}) \odot r. \tag{B.4}$$

Here $\odot$ denotes elementwise multiplication. Both have conditional variance 4 at initialization. Their same-sample gradient increments coincide, giving a common sample-smoothness upper bound. The common centered-noise norm bound $K = 4$ need not be equally tight for the two oracles.

B.3 Configurations of the Synthetic Experiments

Problem seeds and statistical units. The sensitivity experiment uses problem seeds 28924001–28924005 and five sampling seeds per problem, starting at 79523000, 79523100, 79523200, 79523300, and 79523400, respectively, and taking five consecutive integers. Seeds are averaged within a problem before descriptive 95% t intervals are computed over five problem means. Objective-scale transfer retains five existing problem–sampling-seed pairs and reports sample standard deviations. The independent paired-noise confirmation uses problem seeds 28923001–28923005 and twenty consecutive sampling seeds per problem, starting at 79323000, 79323100, 79323200, 79323300, and 79323400, respectively. Repetition counts and statistical units are not pooled across experiments; intervals are not adjusted for multiple comparisons.

Paired noise and independent confirmation. Algorithms do not receive the noise formula and experience conditional variance along their own trajectories. The candidate $\delta = 0.1$ was selected during exploration. Before independent confirmation, we froze the formulas, parameters, 48,001-query budget, and thresholds 0.01, 0.03, and 0.04. Five new problems, twenty seeds each, five methods, and two environments give 1,000 runs. Sections 6.2 and 6.4 use different comparisons from this same batch. The prespecified duration diagnostic requires 10%–90% of records to have variance above 1% of its initial value, followed by a final contiguous low-variance tail. Results appear in Appendix C.1.

Stepsize-scale sensitivity. The seven multipliers in Section 6.3 keep the objective, noise, network, initialization, and query cap fixed. Each method uses its reference configuration without selecting a new center on these problems. Ours, D-NASA, and AdaMDOS scale η, the stepsize multiplier, and γ_{Ada}, respectively. GT-STORM jointly sets $\eta_0 \leftarrow k\eta_0$ and $\rho_{\mathrm{GT}} \leftarrow \rho_{\mathrm{GT}}/k^2$, preserving the refresh schedule; this is a coupled configuration perturbation. Each multiplier uses four methods and five problems with five seeds each, for 700 main trajectories. Problems and sampling seeds are paired across multipliers, the main and proxy streams are independent, and old- and new-point evaluations of one sample are charged separately. At 24,001 queries, Ours, GT-STORM, D-NASA, and AdaMDOS execute 8,000, 12,000, 24,001, and 12,000 updates, respectively.

With GT-STORM's ρ_{GT} strictly fixed, the first momentum coefficient is $\beta_1^{\mathrm{GT}} = 1 - \rho_{\mathrm{GT}}(k\eta_0)^2$. Multipliers $1/8$, $1/4$, $1/2$, and 1 are valid in the current implementation; 2, 4, and 8 yield $\beta_1^{\mathrm{GT}} < 0$ and are not run or counted as numerical divergence. The $k = 1$ runs are reused; the other three valid multipliers add 75 runs.

Objective-scale transfer. We test whether configurations chosen once in the reference environment remain effective after changing the actual objective scale. Four-node rings, dimension 20, and

bounded additive noise $\sigma = 4$ remain fixed, while $c \in \{0.25, 1, 4, 16\}$. Each environment restarts from its corresponding common initialization and receives 24,001 queries per node. The primary metric is M_T; parameter sources are in Appendix B.1.

Constant-noise scan. We fix $c = 1$, the network, problem/seed pairs, and reference configurations, and vary only the constant bounded additive noise level. The original scan uses $\sigma = 0, 1, 4, 16$; a subsequently fixed grid adds $0.1, 0.25, 0.5$ with the same budget and thresholds. These are separate constant-noise runs rather than temporal decay within one run.

Noise-proxy ablation. The main ablation uses 400 trajectories for two environments and two methods from the $\delta = 0.1$ confirmation batch of 1,000 runs. No-proxy removes only the proxy term and independent query. The resulting scale change also changes effective stepsizes and estimator coefficients, which is part of the mechanism under test. Remaining parameters are not retuned. The $\gamma = 0$ variant lies outside the full theorem's $\gamma > 0$ condition.

B.4 Classification Settings

Both datasets use the official 60,000 training and 10,000 test examples, normalized using training statistics, without artificial gradient noise. The 784–64–10 tanh MLP has 50,890 float64 parameters, minibatch size 32, and L2 coefficient 10^{-4}. Each node of the four-node ring assigns weight $1/3$ to itself and each neighbor. Data are partitioned with Dirichlet concentration 0.1.

Dataset-specific parameters retain historical validation selections. Neighbor inherits η, s_0 from the former Global configuration and uses $\gamma = c_z/4$, where c_z is the noise-proxy coefficient stored in that historical configuration. Table 4 reports the effective γ used by the current implementation. We neither tune on the present test results nor replace classification settings with the synthetic configurations.

Table 4: Frozen classification configurations.

Method	Fashion-MNIST	KMNIST
D-SGD	Stepsize 0.1	Stepsize 0.24
GT	Stepsize 0.0625	Stepsize 0.1
D-NASA	Stepsize multiplier 3	Stepsize multiplier 3
AdaMDOS	$\gamma_{\mathrm{Ada}} = 0.002,\ \rho_{\mathrm{Ada}} = 0.001$ $\eta = 1.25,\ s_0 = 1.5$	$\gamma_{\mathrm{Ada}} = 0.002,\ \rho_{\mathrm{Ada}} = 0.001$ $\eta = 2,\ s_0 = 1$
Ours	$\gamma = 0.02$	$\gamma = 0.005$

AdaMDOS uses relaxation, momentum, and second-moment decay 0.9. D-SGD is ordinary fixed-stepsize decentralized stochastic gradient descent; GT is ordinary stochastic gradient tracking, not GT-STORM. D-NASA uses normalized directions and its decay rule; AdaMDOS uses second-moment preconditioning and same-sample two-point evaluations. Query counts include initialization and repeated evaluations at old points.

Neighbor accuracy was rerun using seeds 105–109 on both datasets (ten trajectories). The forty accuracy records for the four unchanged baselines are reused from archived results with their source hashes retained. Timing was remeasured for all five methods using seed 105 and five repeats. Accuracy variation therefore reflects training seeds, whereas timing variation reflects controlled repeated runs. This is not a fresh-seed independent classification confirmation. Test accuracy is computed offline on the parameter-averaged model; that averaging does not enter training updates. Historical Global equal-update comparisons, no-STORM/no-tracking ablations, and CIFAR diagnostics are not results for the current Neighbor method.

B.5 Communication and Timing Protocol

The controlled measurements in Table 1 use a single-host four-process CPU/Gloo implementation, with seed 105 and five repeats per method. Accuracy and timing repetitions differ as described in Appendix B.4; operation counts and timing breakdowns are in Appendix C.5.

All methods perform zero global reductions within training updates. Vector payload is the number of vector mixes times two neighbors times 50,890 parameters times eight bytes, divided by 2^{30} for GiB. The common architecture and budget give the same payload on both datasets. GT and D-NASA can pack several vectors from one round into a message, so vector message counts differ from vector-mix counts.

Ours performs 2,000 updates. Each node i piggybacks its envelope Q_t^i on the tracking vector for use in the next round. Each node sends 4,000 extra scalar values across its two neighbors: 32,000 bytes, or 31.25 KiB, without scalar-only messages. Payload excludes protocol headers and lower-level wire overhead. Since scalars and vectors share packets, scalar-processing time is not isolated from the communication region.

Each run's breakdown uses the same process that has the largest total time, rather than combining maxima from different processes. The communication region includes waiting, buffers, and local mixing, not only network transmission; component standard deviations are not additive. Initialization, all training updates, and scalar handling are timed. Process startup, loading, warm-up, evaluation, validation checks, saving, and boundary barriers are excluded.

The initial KMNIST batch passed complete-training checks for all methods, but the first formal Neighbor timing attempt hit a 600-second Gloo receive timeout. After retaining the failed attempt, we reran the entire KMNIST batch with a new process group and output directory, preserving algorithms, communication code, parameters, seed, order, and thresholds. There are fifty successful formal timing records and one failed attempt. Means exclude failure waiting and recovery costs and do not establish end-to-end reliability or multi-host performance.

C Supplementary Experimental Results and Analysis

Results are grouped by experimental question; settings and reproducibility details are collected in Appendix B.

C.1 Paired Noise and Proxy Ablation

Sections 6.2 and 6.4 share the independent confirmation batch of 1,000 runs, with no numerical failures. The construction and frozen protocol appear in Appendices B.2–B.3.

Table 5: Per-node query costs to prespecified stationarity thresholds under vanishing noise.

Stationarity threshold	Ours	GT-STORM	D-NASA	AdaMDOS
$M_T \le 0.01$	**3,262**	$8,569$	$7,869$	$12,953$
$M_T \le 0.03$	**1,090**	$2,849$	$2,135$	$4,319$
$M_T \le 0.04$	**817**	$2,129$	$1,541$	$3,239$

Entries are medians of per-run first-hit counts. All entries have 100/100 successful runs. Under persistent noise, no method reaches $M_T \le 0.01$ within the budget; see Appendix C.1.

Table 6: Median first-hit query counts and numbers of successful runs under persistent noise.

Threshold	Ours	GT-STORM	D-NASA	AdaMDOS	No-proxy
0.01	$> 48,001$ (0/100)	$> 48,001$ (0/100)	$> 48,001$ (0/100)	$> 48,001$ (0/100)	$> 48,001$ (0/100)
0.03	$8,140$ (100/100)	$7,979$ (100/100)	$16,529$ (100/100)	$> 48,001$ (0/100)	$> 48,001$ (0/100)
0.04	$3,829$ (100/100)	$4,743$ (100/100)	$6,817$ (100/100)	$> 48,001$ (0/100)	$> 48,001$ (0/100)

Under vanishing noise, all four full methods reach all three thresholds in 100/100 runs; their median query counts are in Table 5. No-proxy requires 4,575, 1,527, and 1,145 queries for thresholds 0.01, 0.03, and 0.04, respectively, also with 100/100 successes. Parentheses in Table 6 give successes out

of 100; $> 48{,}001$ denotes a median right-censored at the query cap with unsuccessful runs included. Under persistent noise, Ours reaches 0.04 earlier than GT-STORM but reaches 0.03 later and has a higher final M_T, illustrating differences across stationarity thresholds.

For Ours under vanishing noise, high-variance records constitute about 0.578%, and the median query count at the start of the final low-variance tail is 283. None of the 100 runs meets the extended-stochastic-phase criterion in Appendix B.3. Thus, this confirmation mainly demonstrates finite-budget gains after early noise decay.

Supplementary proxy results. This ablation tests the proxy under frozen configurations; No-proxy inherits the remaining parameters and is not independently tuned. Exploratory results show that proxy gains depend on noise conditions: in historical zero-noise runs the two variants have identical equal-update trajectories, with extra query cost for Full; at $\delta = 0.03$, Ours does not beat every baseline, and at $\delta = 0.3$, No-proxy is better. These exploratory settings have not received new independent confirmation and are reported separately from the $\delta = 0.1$ confirmation.

C.2 Complete Stepsize-Scale Sensitivity Results

We report absolute values and descriptive intervals for all seven multipliers in Section 6.3, followed by the GT-STORM coupling diagnostic. Configurations are in Appendix B.3.

Table 7: Final mean M_T and descriptive 95% intervals for all stepsize multipliers.

k	Ours	GT-STORM*	D-NASA	AdaMDOS
$1/8$	0.032308 $[0.031736, 0.032879]$	0.018787 $[0.018558, 0.019017]$	0.712844 $[0.710553, 0.715135]$	0.087300 $[0.087084, 0.087516]$
$1/4$	0.023973 $[0.023474, 0.024472]$	0.016201 $[0.015980, 0.016423]$	0.134996 $[0.134349, 0.135643]$	0.059885 $[0.059651, 0.060120]$
$1/2$	0.022236 $[0.021874, 0.022597]$	0.015507 $[0.015278, 0.015735]$	0.032422 $[0.032286, 0.032558]$	0.053817 $[0.053540, 0.054093]$
1	0.021898 $[0.021612, 0.022183]$	0.015860 $[0.015731, 0.015989]$	0.028240 $[0.028091, 0.028389]$	0.061675 $[0.061343, 0.062007]$
2	0.021909 $[0.021660, 0.022159]$	0.026518 $[0.025793, 0.027242]$	0.037179 $[0.036957, 0.037401]$	0.081382 $[0.081021, 0.081743]$
4	0.022227 $[0.021972, 0.022481]$	0.142893 $[0.134844, 0.150943]$	0.052008 $[0.051722, 0.052295]$	0.114620 $[0.114315, 0.114925]$
8	0.023153 $[0.022901, 0.023404]$	5.919×10^{15} $[5.129 \times 10^{15}, 6.709 \times 10^{15}]$	0.073773 $[0.073520, 0.074025]$	0.167019 $[0.166843, 0.167195]$

We first average the five sampling seeds within each problem, then compute intervals using the five problem means as statistical units. The asterisk marks GT-STORM's coupled adjustment. Very large values use scientific notation; intervals are pointwise descriptive intervals without multiple-comparison correction.

Table 8: GT-STORM's coupled-parameter diagnostic.

k	Coupled M_T	Fixed-ρ_{GT} M_T	Fixed-ρ_{GT} β_1^{GT}
$1/8$	0.018787	0.449089	0.995133
$1/4$	0.016201	0.228874	0.980531
$1/2$	0.015507	0.027208	0.922125
1	0.015860	0.015860	0.688502
2	0.026518	Invalid; not run	-0.245994
4	0.142893	Invalid; not run	-3.983974
8	5.919×10^{15}	Invalid; not run	-18.935897

None of the 775 executed trajectories contains NaN/Inf. For coupled GT-STORM at $k = 8$, the mean full-trajectory M_T is approximately 5.92×10^{15}, with individual values spanning about $4.27\times$

10^{15}–7.46×10^{15}. Post-hoc inspection shows severe early overshoot, whereas the final normalized instantaneous gradient is about 0.118–0.364. We record this as finite-valued early overshoot, distinct from NaN/Inf failure. One full formal trajectory was checked against the original implementation. On the current objective family, noise level, and seven-point grid, Ours is relatively insensitive to upward stepsize scaling; GT-STORM is better at the reference and smaller multipliers.

C.3 Objective-Scale Transfer

This experiment evaluates objective-scale transfer with frozen configurations; settings are in Appendix B.3.

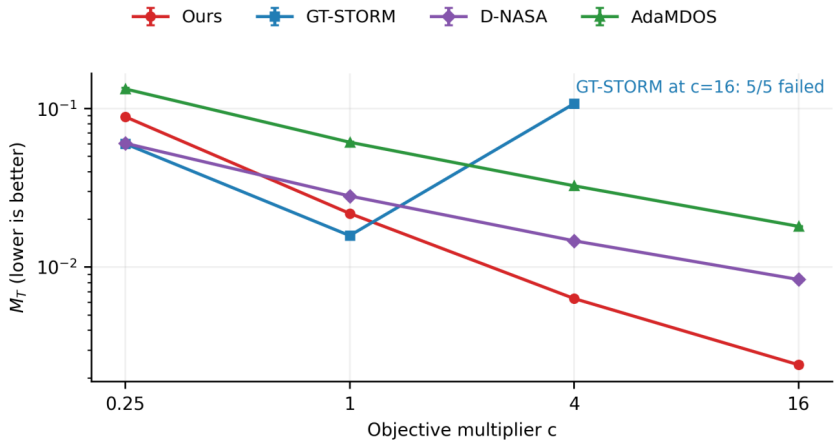


Figure 4: Frozen-configuration objective-scale transfer. Points and error bars are means and sample standard deviations over five paired problems. GT-STORM has 5/5 numerical failures at $c = 16$; no finite point is drawn.

At $c = 4$, Ours has mean $M_T = 0.006328$, below all three baselines. At $c = 16$, its 0.002419 is below D-NASA and AdaMDOS, while all five GT-STORM runs fail numerically. At $c = 0.25$ and $c = 1$, GT-STORM is better. Objective scaling jointly changes gradient magnitude and relative noise strength, so this experiment examines frozen-configuration transfer under that joint change. Tables 9 and 10 report normalized endpoints and raw ergodic gradient norms at 24,001 queries, respectively, as means and sample standard deviations over five paired instances.

Table 9: Normalized objective-scale transfer endpoints.

Method	$c = 0.25$	$c = 1$	$c = 4$	$c = 16$
Ours	0.088538 ± 0.002474	0.021771 ± 0.000497	0.006328 ± 0.000134	0.002419 ± 0.000096
GT-STORM	0.059843 ± 0.003233	0.015779 ± 0.000483	0.107122 ± 0.004980	5/5 failed
D-NASA	0.060256 ± 0.001776	0.028036 ± 0.000334	0.014631 ± 0.000047	0.008344 ± 0.000019
AdaMDOS	0.132625 ± 0.002980	0.061190 ± 0.000548	0.032504 ± 0.000075	0.018046 ± 0.000135

Table 10: Unnormalized objective-scale transfer endpoints.

Method	$c = 0.25$	$c = 1$	$c = 4$	$c = 16$
Ours	0.080191 ± 0.002837	0.078874 ± 0.002419	0.091707 ± 0.003148	0.140281 ± 0.007742
GT-STORM	0.054166 ± 0.002283	0.057143 ± 0.001080	1.553132 ± 0.096869	5/5 failed
D-NASA	0.054567 ± 0.001667	0.101562 ± 0.002043	0.212014 ± 0.003870	0.483615 ± 0.007705
AdaMDOS	0.120112 ± 0.003360	0.221663 ± 0.003944	0.470978 ± 0.006943	1.045943 ± 0.018126

The normalized endpoint falls with increasing c, whereas the raw endpoint does not follow the same trend; cross-scale comparisons should use both reporting units. Neighbor completes finite-valued computations in all twenty scale-transfer runs.

C.4 Constant-Noise Scan and Nonzero Low-Noise Results

We report seven noise levels at 48,001 queries per node to compare optimization quality across noise strengths. Each level is a separate constant-noise run; settings are in Appendices B.2–B.3.

Table 11: Mean final M_T in the constant-noise scan.

σ	Ours	GT-STORM	D-NASA	AdaMDOS
16	0.1200397	0.0469886	0.0478556	0.1218345
4	0.0172172	0.0120669	0.0228868	0.0582413
1	0.0080446	0.0032720	0.0117520	0.0298819
0.5	0.0056102	0.0017417	0.0085965	0.0213841
0.25	0.0038444	0.0009697	0.0063947	0.0150000
0.1	0.0021728	0.0005155	0.0044850	0.0091142
0	0.0001556	0.0002335	0.0024594	0.0022372

As σ decreases from 1 to 0.1, Ours improves from approximately 0.008045 to 0.002173, but remains above GT-STORM at every tested nonzero noise level. Other methods also benefit from lower noise; these results supplement the relationship between noise strength and finite-budget optimization quality.

C.5 Communication Counts and Training-Time Breakdown

We give communication/update counts and timing breakdowns corresponding to Table 1. Accounting rules, timed regions, and failed-batch handling appear in Appendix B.5.

Table 12: Communication and update counts at 6,001 queries per node.

Method	Updates	Vector mixes	Vector messages per node	Scalar values per node	Scalar bytes per node
D-SGD	6000	6000	12000	0	0
GT	6000	12000	12000	0	0
D-NASA	6001	18003	12002	0	0
AdaMDOS	3000	6001	12002	0	0
Ours	2000	4001	8002	4000	32000

Cumulative vector payload. A common query budget permits different update counts. Ours uses three queries per update and completes 2,000 updates and 4,001 vector mixes at 6,001 queries per node. D-SGD, GT, D-NASA, and AdaMDOS perform 6,000, 12,000, 18,003, and 6,001 vector mixes, respectively. Cumulative traffic is determined jointly by update counts within the budget and each method's vector-exchange rule.

Controlled training time. On both datasets, the main timing differences occur in the neighbor-communication region. Fewer cumulative vector exchanges, together with the corresponding calls and local processing, are consistent with Ours having a lower total training time.

Table 13: Controlled training-time breakdown on both datasets.

Dataset	Method	Gradient (s)	Neighbor region (s)	Other (s)
Fashion-MNIST	D-SGD	11.58 ± 1.13	41.66 ± 6.95	3.98 ± 0.29
Fashion-MNIST	GT	10.67 ± 1.61	73.81 ± 13.87	4.63 ± 0.41
Fashion-MNIST	D-NASA	10.80 ± 1.65	104.57 ± 19.48	15.25 ± 2.57
Fashion-MNIST	AdaMDOS	10.15 ± 1.46	38.47 ± 6.18	5.42 ± 0.44
Fashion-MNIST	Ours	10.12 ± 1.63	28.52 ± 7.08	3.72 ± 0.29
KMNIST	D-SGD	8.39 ± 0.38	30.63 ± 0.43	3.00 ± 0.10
KMNIST	GT	9.70 ± 0.15	69.47 ± 0.80	3.89 ± 0.05
KMNIST	D-NASA	8.79 ± 0.15	92.39 ± 0.84	13.32 ± 0.57
KMNIST	AdaMDOS	7.68 ± 0.42	28.85 ± 1.28	4.08 ± 0.09
KMNIST	Ours	7.67 ± 0.31	21.70 ± 0.87	2.79 ± 0.06

Entries are mean $\pm$ sample standard deviation in seconds over five fixed-seed repeats. The neighbor region includes waiting, buffers, and local mixing, not just network transmission. See Appendix B.5 for the full timing protocol.

D Relationship to Existing Analyses

This appendix explains the proof interfaces specific to Algorithm 1. Recursive STORM estimation, gradient tracking, and spectral contraction are established ingredients (Cutkosky & Orabona, 2019; Zhang et al., 2021; Huang & Zhao, 2024). Adaptive recursive estimation has also been studied centrally (Levy et al., 2021; Liu et al., 2022), and D-NASA addresses problem-parameter-free decentralized optimization (Li et al., 2025). The present analysis does not claim these ingredients individually as new. Its object is their coupling with the actual nodewise direction-and-noise scales and the resulting cumulative-variance guarantee.

D.1 The average update with nodewise stepsizes

Double stochasticity preserves $\bar{w}_t = \bar{u}_t$, but unequal local stepsizes mean that the average-model update is not simply $\bar{x}_{t+1} = \bar{x}_t - \bar{\alpha}_t \bar{u}_t$. Instead,

$$\bar{x}_{t+1} = \bar{x}_t - \bar{\alpha}_t \bar{u}_t - r_t, \qquad r_t = \frac{1}{n}\sum_i \alpha_t^i (w_t^i - \bar{u}_t). \tag{D.1}$$

With $D_t = n^{-1}\sum_i \|w_t^i - \bar{u}_t\|^2$ and $a_{\max} = \eta/s_0$, weighted Cauchy–Schwarz gives $\|r_t\|^2/\bar{\alpha}_t \le a_{\max} D_t$. Furthermore, if $g_t^{\mathrm{loc}} = n^{-1}\sum_i \nabla f_i(x_t^i)$ and $G_t = \nabla F(\bar{x}_t)$, then

$$\bar{u}_t - G_t = \bar{e}_t + (g_t^{\mathrm{loc}} - G_t), \qquad \|g_t^{\mathrm{loc}} - G_t\| \le L\sqrt{C_t}. \tag{D.2}$$

Thus tracking disagreement, estimation error, and model disagreement must all be controlled for the actual adaptive coefficients.

D.2 Why local error control needs a scale comparison

The estimator recursion controls the locally weighted quantity $Z_t = n^{-1}\sum_i \beta_t^i \|e_t^i\|^2$, whereas average-model descent requires $\bar{\alpha}_t\|\bar{e}_t\|^2$. Lemma A.8 supplies the conversion

$$\bar{\alpha}_t\|\bar{e}_t\|^2 \le \chi_t a_{\max} Z_t, \qquad \chi_t = \left(\frac{1}{n}\sum_i s_t^i\right)\left(\frac{1}{n}\sum_i \frac{1}{s_t^i}\right) \le \frac{(\kappa+1)^2}{4\kappa}. \tag{D.3}$$

No independence between errors and scales is used.

Importantly, the uniform ratio bound is proved rather than assumed. Local memory makes the envelopes nondecreasing. Normalized model increments and bounded centered noise then bound estimator and tracker increments. Finite-hop propagation yields the scale comparison in Lemma A.4 before any expected descent or cumulative-budget closure is invoked. The resulting κ may be large and depends on the graph diameter; it establishes bounded multiplicative mismatch, not synchronization or closeness to one.

D.3 Predictable noise budgets and raw-budget closure

The proxy shares one evaluation with the estimator, but the refresh coefficient is chosen before the fresh pair. Conditional proxy unbiasedness therefore controls the predictable weighted variance without assuming proxy–estimator independence (Lemma A.5).

Let $\mathcal{B}_t = n^{-1}\sum_i B_t^i$, $h_T = \mathbb{E}\log(\mathcal{B}_T/s_0^2)$, and $y_T = \mathbb{E}\sqrt{\mathcal{B}_T}$. The received envelope is not inserted into the local raw-budget accumulator. The pathwise relation

$$Q_t^i \le \max_j (B_t^j + \|w_t^j\|^2) \le n(\mathcal{B}_t + A_t) \tag{D.4}$$

connects envelope-weighted descent to raw-budget growth. Lemma A.10 derives the pathwise decomposition without the scale-ratio bound. The expected closure subsequently uses the already established descent constants:

$$y_T \le \sqrt{s_0^2 + \gamma \mathcal{V}_T} + d_0 + d_1 h_T, \qquad h_T \le 2\log(y_T/s_0). \tag{D.5}$$

The constants and explicit solution are given in (A.53)–(A.55). This ordering avoids a circular dependence between scale comparison and budget closure. Logarithmic telescoping, Jensen's inequality, and Cauchy–Schwarz remain standard tools; the proof supplies the budgets needed to apply them to this particular recursion.

D.4 SCOPE AND COMPARISON UNITS

The convergence theorem concerns $\mathbb{E}\|\nabla F(x_{\widehat{t}}^{I})\|$ at an independent uniformly selected local iterate. It is neither a squared-gradient guarantee nor a last-iterate rate. Its cumulative expected conditional variance is defined along this algorithm's own trajectory and cannot be numerically identified with another method's trajectory budget.

The analysis assumes same-sample mean-square smoothness, almost-sure bounded centered noise, fresh independent proxy samples, and a fixed synchronous network. The constant retains dependence on L, K, graph parameters, and initialization, even though the update does not take their numerical values as inputs. In particular, this result does not cover arbitrary asynchronous delays or general unbounded finite-variance noise.

Comparisons must match assumptions, output rules, stationarity metrics, and cost units. Each update here uses three oracle evaluations from two fresh samples and sends a scalar envelope to each neighbor in addition to the vector mixes. Cumulative-variance dependence does not by itself establish novelty or dominance over existing adaptive methods, and the current theorem makes no universal finite-budget performance claim.