SPADE-DFL: COMMUNICATION-EFFICIENT DECENTRALIZED FEDERATED LEARNING VIA DERIVATIVE-FREE LINEARIZED ADMM

Anonymous authors
Paper under double-blind review

ABSTRACT

Reducing communication in derivative-free decentralized learning requires controlling the disagreement accumulated over multiple local updates. This paper develops SPADE-DFL, a primal–dual method that allows the number of local function-value updates between neighbor exchanges to grow with the computation budget while preserving the nonprivate convergence order. For smooth nonconvex objectives under uniform query-moment bounds, the prescribed nonprivate schedule achieves a time-averaged stationarity and consensus bound of $\mathcal{O}(T^{-1/3})$ using only $\Theta(T^{2/3})$ communication rounds, where T is the number of local updates per client. For private training, the accumulated data-dependent increment is isolated from the graph correction, allowing one protected state per client and round to generate all outgoing messages. We prove client-level differential privacy for the full interactive transcript and quantify the resulting optimization error over a finite horizon. Experiments on four classification tasks show that SPADE-DFL achieves higher mean test accuracy than existing decentralized learning methods. Code is available at https://anonymous.4open.science/r/SPADE-DFL-1E2D.

1 INTRODUCTION

Decentralized federated learning (DFL) trains a shared model through neighbor exchanges while keeping data local (Zehtabi et al., 2025). Communication constraints motivate clients to perform several local updates before exchanging model information (Wu et al., 2025; Alghunaim, 2024). The central question is how much communication local computation can replace without compromising progress toward the shared objective (Ren et al., 2026a).

When gradients are unavailable, local computation relies on randomized function evaluations (Ghadimi & Lan, 2013; Ye et al., 2025; Huang et al., 2024). Single-point methods extend this information model to decentralized learning (Mhanna & Assaad, 2023; 2024). Local models can nevertheless drift apart during training, so additional queries offer progress at the cost of a growing coordination problem. Compression can reduce the information transmitted in an exchange, but the number of exchanges remains part of the communication cost (Song et al., 2022; Nassif et al., 2025; He et al., 2023). The question is therefore more specific. *How much communication can local loss evaluations replace while preserving the convergence order of collective learning?*

Local computation reduces the frequency of neighbor exchanges but also permits local models to drift apart before the next exchange. With single-point function-value information, the smoothing radius affects the approximation error of the local direction. The local training length, stepsize, smoothing radius and network gain therefore need to be selected jointly. Their interaction determines whether reducing the exchange frequency also reduces the total communication required to reach a given optimization accuracy.

Privacy adds to this dependence on communicated states. Reconstruction attacks show that model information can reveal local training data (Guo et al., 2026). Distributed privacy mechanisms exploit graph structure (Rizk et al., 2023; Allouah et al., 2024), including formulations based on noisy ADMM iterations (Cyffers et al., 2023). Each released state summarizes a local query sequence whose influence continues through subsequent neighbor updates. Its protective perturbation therefore alters the feedback driving later computation. Privacy analysis follows this influence through the complete interaction using composition of successive releases (Bun & Steinke, 2016; Mironov, 2017).

SPADE-DFL separates accumulated single-point updates from the graph correction fixed by preceding exchanges, yielding three contributions.

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.

1. **Communication savings from local loss evaluations.** For smooth nonconvex objectives under uniform query-moment bounds, $\tau = \Theta(T^{1/3})$ local steps per exchange preserve the $\mathcal{O}(T^{-1/3})$ time-averaged stationarity–consensus rate under the prescribed nonprivate schedule. For joint tolerance $\varepsilon_{\mathrm{stat}}$, sufficient rounds decrease from $\mathcal{O}(\varepsilon_{\mathrm{stat}}^{-3})$ for the $\tau = 1$ specialization to $\mathcal{O}(\varepsilon_{\mathrm{stat}}^{-2})$.
2. **Single-point local training and network dynamics.** The component memory preserves fresh estimates' conditional mean with a controlled second moment. An exact message recursion characterizes homogeneous network stability and the admissible round gain under synchronized updates.
3. **Privacy across the full interaction.** One release per client and round gives differential privacy for the full interactive transcript under replacement of a fixed-size client dataset; a finite-horizon bound quantifies clipping, Gaussian perturbations and local drift.

2 Related Work

2.1 Communication in Decentralized Learning

Local training reduces exchanges but must control disagreement from heterogeneous objectives (Wu et al., 2025). ProxSkip accelerates communication through randomized synchronization for strongly convex problems (Mishchenko et al., 2022); local exact diffusion adds bias correction (Alghunaim, 2024), and local training ADMM freezes neighbor corrections over several stochastic updates (Ren et al., 2026a). Its private variant clips and perturbs gradients at each local step, with privacy under single-record addition or removal (Ren et al., 2026b). Communication also depends on exact consensus schedules (Ding et al., 2023), sparse neighbor interactions (You & Pu, 2024), heterogeneity correction (Yuan et al., 2023), aggregation weights (Zhai et al., 2026), and decentralized training stability (Sun et al., 2026). Compression reduces message payloads (He et al., 2023) through directed gradient tracking (Song et al., 2022), differential error feedback (Nassif et al., 2025), BEER's gradient tracking (Zhao et al., 2022), or MoTEF's momentum tracking (Islamov et al., 2025); related online methods analyze regret for time-varying objectives (Li et al., 2024a).

2.2 Learning from Function Values

Zeroth-order optimization estimates directions from function evaluations (Ghadimi & Lan, 2013), including one noisy query per update in single-point decentralized methods (Mhanna & Assaad, 2023; 2024). Curvature information (Ye et al., 2025), stochastic ADMM (Huang et al., 2024), compression (Hua et al., 2026), and quantized gradient tracking under the Polyak Łojasiewicz condition (Xu et al., 2024) address estimation or communication costs. FedZO combines function evaluations with local training (Fang et al., 2022), whereas DeComFL removes model-dimension dependence from the communication payload (Li et al., 2025). MeZO uses paired forward evaluations for language model adaptation (Malladi et al., 2023), and its SVRG extension reuses reference information (Gautam et al., 2024). For decentralized solvers, ADMM enforces neighbor agreement (Boyd et al., 2011; Shi et al., 2014); linearization yields explicit primal updates (Ling et al., 2015), while curvature-aided methods avoid inner communication loops for convex composite objectives (Li et al., 2023).

2.3 Privacy across Distributed Interactions

Distributed privacy mechanisms use graph homomorphic noise (Rizk et al., 2023), quantized subspace perturbations (Li et al., 2024b), or positive incentive noise (Wang et al., 2026). Muffliato studies amplification under pairwise network observation (Cyffers et al., 2022), whereas DECOR cancels Gaussian perturbations across neighbors using shared secret randomness (Allouah et al., 2024). Private ADMM uses noisy fixed-point iterations (Cyffers et al., 2023), with concentrated and Rényi privacy providing composition rules (Bun & Steinke, 2016; Mironov, 2017). Clipping can introduce persistent bias (Koloskova et al., 2023); DP-FedSAM studies private local training with sharpness-aware optimization and sparsification (Shi et al., 2025). In zeroth-order learning, DPZero supports private adaptation without backpropagation (Zhang et al., 2024), while PAZO uses public data under a similarity assumption (Gong & Li, 2025).

SPADE-DFL combines component-memory single-point estimation with local training ADMM. Under uniform query-moment bounds, its prescribed nonprivate schedule permits longer local stages while preserving the stationarity and consensus convergence order. Clipping and perturbing the accumulated data-dependent increment once per round establishes client-level privacy for the full interactive transcript under replacement of an entire fixed-size local dataset. Appendix A provides the complete related work, with algorithmic features compared in Table 3.

3 Problem Formulation and Methodology

3.1 Problem Formulation

Consider a decentralized empirical-risk minimization system in which training samples remain distributed across $N \geq 2$ clients and no central server has direct access to their union (McMahan et al., 2017; Kairouz et al., 2021). The N clients are connected by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, N\}$ and $\mathcal{E}$ is the edge set. Client i communicates only with neighbors in $\mathcal{N}_i := \{j : \{i, j\} \in \mathcal{E}\}$ and stores $\mathcal{D}_i = \{\xi_{i,h}\}_{h=1}^{m_i}$, where the sample $\xi_{i,h}$ induces the component loss $f_{i,h} : \mathbb{R}^d \to \mathbb{R}$. The client-specific empirical risk and the client-uniform learning objective are defined respectively as

$$f_i(x) = \frac{1}{m_i} \sum_{h=1}^{m_i} f_{i,h}(x), \qquad F(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x), \tag{1}$$

which separates the within-client empirical average from the network-level aggregation. Data heterogeneity and the local sample size are absorbed into f_i. The component losses $f_{i,h}$, local empirical risks f_i and aggregate objective F may be nonconvex. At a queried model point, client i observes component loss values but has no access to $\nabla f_{i,h}$ or ∇f_i. This information pattern arises when the local objective is exposed through the external black-box interface or its derivatives are computationally prohibitive (Mhanna & Assaad, 2023; Huang et al., 2024; Ye et al., 2025).

To implement the common-model objective through neighbor exchanges, assign a local replica x_i to each client. The resulting consensus formulation can be obtained from (Boyd et al., 2011) as

$$\min_{\{x_i\}_{i=1}^N} \quad \frac{1}{N} \sum_{i=1}^{N} f_i(x_i) \quad \text{s.t.} \quad x_i = x_j, \quad \{i, j\} \in \mathcal{E}. \tag{2}$$

Since (2) separates local objectives and couples their replicas only along graph edges, connectivity makes the edge constraints equivalent to $x_1 = \cdots = x_N$, thereby recovering $\min_x F(x)$ and enabling decentralized ADMM through local loss evaluations and neighbor messages. The required objective regularity and network conditions are stated in Assumptions 1 and 2, respectively.

Assumption 1 ((Ghadimi & Lan, 2013; Mhanna & Assaad, 2023))**.** *Each component loss is continuously differentiable and L_f-smooth, i.e., $\|\nabla f_{i,h}(x) - \nabla f_{i,h}(y)\| \leq L_f \|x - y\|$, $\forall x, y \in \mathbb{R}^d$. Furthermore, F is bounded below by $F_\star > -\infty$.*

Assumption 2 ((Shi et al., 2014))**.** *The communication graph $\mathcal{G}$ is fixed, undirected, unweighted, and connected. Messages are exchanged synchronously over its edges.*

For the graph representation, orient each edge arbitrarily and let $B \in \mathbb{R}^{N \times |\mathcal{E}|}$ be the resulting incidence matrix. For an edge $e = (i, j)$, set $B_{i,e} = 1$, $B_{j,e} = -1$, and all other entries in column e to zero. Then $L_{\mathcal{G}} = BB^\top$ and $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_N$ define the graph Laplacian and its ordered eigenvalues, respectively. Suppressing unambiguous Kronecker products with I_d, denote $\Pi = \left(I_N - \frac{1}{N}\mathbf{1}\mathbf{1}^\top\right) \otimes I_d$. The local recursion uses a memory-based direction formed from single-point queries.

3.2 Single-Point Estimation and Memory Properties

At a point x measurable with respect to the pre-query history $\mathcal{F}$, client i observes $\widetilde{f}_{i,h}(x + \mu u) = f_{i,h}(x + \mu u) + \zeta$. Here, $\mu > 0$, $u \in \mathbb{R}^d$, and ζ denote the smoothing radius, query direction, and noise, respectively.

Assumption 3 ((Mhanna & Assaad, 2023; 2024))**.** *For every generated query, constants $c_u, R_u, M_2 > 0$ satisfy $\mathbb{E}[u \mid \mathcal{F}] = 0$, $\mathbb{E}[uu^\top \mid \mathcal{F}] = c_u I_d$, and $\|u\| \leq R_u$ almost surely. Moreover, $\mathbb{E}[\zeta \mid \mathcal{F}, u] = 0$ and $\mathbb{E}[|\widetilde{f}_{i,h}(x + \mu u)|^2 \mid \mathcal{F}] \leq M_2$.*

The unnormalized single-point direction uses one function value:

$$q_{i,h}(x; u, \zeta) = \frac{1}{c_u} u \widetilde{f}_{i,h}(x + \mu u). \tag{3}$$

At local step (k, t), $q_{i,h,k}^t$ denotes this query at $\phi_{i,k}^t$ with radius μ_k. Each round starts from $\phi_{i,k}^0 = x_{i,k}$ and initializes one memory entry per component, setting $r_{i,h,k}^0 = x_{i,k}$ and

$$a_{i,h,k}^0 = \frac{1}{c_u} u_{i,h,k}^{\mathrm{ref},0} \widetilde{f}_{i,h}\left(r_{i,h,k}^0 + \mu_k u_{i,h,k}^{\mathrm{ref},0}\right). \tag{4}$$

Let $\mathcal{F}_{k,t}$ denote the pre-step history, including the current iterates and memory but excluding current batches, directions, and query noises. Conditional on this history, sample $\mathcal{B}_{i,k}^t$ uniformly with

size $1 \le b_i \le m_i$, independently of current query randomization and across clients. With $\bar{a}_{i,k}^t = m_i^{-1} \sum_h a_{i,h,k}^t$, form

$$v_{i,k}^t = \frac{1}{b_i} \sum_{h \in \mathcal{B}_{i,k}^t} \left(q_{i,h,k}^t - a_{i,h,k}^t \right) + \bar{a}_{i,k}^t. \tag{5}$$

Then replace each sampled pair by $(r_{i,h,k}^{t+1}, a_{i,h,k}^{t+1}) = (\phi_{i,k}^t, q_{i,h,k}^t)$, retain unsampled entries, and recompute $\bar{a}_{i,k}^{t+1}$.

Lemma 1. *Under Assumptions 1 and 3, set $c_r = L_f R_u^3/(2c_u)$ and $Q^2 = R_u^2 M_2/c_u^2$.* (i) *Each component query satisfies $\mathbb{E}[q_{i,h}(x;u,\zeta) \mid \mathcal{F}] = \mu \nabla f_{i,h}(x) + r_{i,h}(x,\mu)$, with $\|r_{i,h}(x,\mu)\| \le c_r \mu^2$ and $\mathbb{E}[\|q_{i,h}(x;u,\zeta)\|^2] \le Q^2$.* (ii) *The memory direction satisfies $\mathbb{E}[v_{i,k}^t \mid \mathcal{F}_{k,t}] = \mu_k \nabla f_i(\phi_{i,k}^t) + r_{i,k}^t$, with $\|r_{i,k}^t\| \le c_r \mu_k^2$, $m_i^{-1} \sum_h \mathbb{E}[\|a_{i,h,k}^t\|^2] \le Q^2$, and $\mathbb{E}[\|v_{i,k}^t\|^2] \le 9Q^2$.*

Thus, the memory correction preserves the conditional mean of fresh queries with a controlled second moment. Appendix B contains the complete formulation and supporting example.

3.3 SPADE-DFL

Use public, data-independent initialization $x_{i,0} = x_0$, $z_{ij,0} = \rho x_0$, graph and schedules, with $\eta_k, \mu_k, \rho, \beta > 0$. Local randomization is independent across clients; release noise is independent of pre-release variables and conditionally independent across clients and rounds.

3.3.1 Local Derivative-Free Training

At round k, set $\phi_{i,k}^0 = x_{i,k}$ and freeze

$$p_{i,k} = \rho |\mathcal{N}_i| x_{i,k} - \sum_{j \in \mathcal{N}_i} z_{ij,k}. \tag{6}$$

For $t = 0, \dots, \tau_i - 1$, update

$$\phi_{i,k}^{t+1} = \phi_{i,k}^t - \eta_k \left(v_{i,k}^t + \mu_k \beta p_{i,k} \right). \tag{7}$$

The unreleased state is $y_{i,k+1} := \phi_{i,k}^{\tau_i}$; the accumulated data-dependent increment is $s_{i,k} := -\eta_k \sum_{t=0}^{\tau_i - 1} v_{i,k}^t$.

3.3.2 Private Release and Decentralized Communication

Using the clipping and Gaussian noise in Algorithm 1, release

$$x_{i,k+1} = x_{i,k} - \tau_i \eta_k \mu_k \beta p_{i,k} + \bar{s}_{i,k} + \nu_{i,k}. \tag{8}$$

The clipping error is $e_{i,k}^{\mathrm{cl}} := \bar{s}_{i,k} - s_{i,k}$. Each outgoing message uses the same released state:

$$m_{i \to j,k} = z_{ij,k} - 2\rho x_{i,k+1}, j \in \mathcal{N}_i. \tag{9}$$

Upon receiving $m_{j \to i,k}$, client i updates

$$z_{ij,k+1} = \frac{1}{2} \left(z_{ij,k} - m_{j \to i,k} \right). \tag{10}$$

This uses one d-dimensional message per directed edge and round (Fig. 1). Equivalent updates are given in Appendix G.2.

4 Theoretical Analysis

The exact message recursion determines the graph correction from the released history. Conditioning on this history gives the release law used for privacy accounting. The disagreement and descent bounds then yield the communication and query budgets.

4.1 Exact Message Recursion and Transcript Privacy

Variables without node or edge subscripts denote stacked vectors and Kronecker products with I_d are omitted when unambiguous. For each oriented edge $e = (i,j)$, define $\omega_{e,k} := (z_{ij,k} - z_{ji,k})/2$. Set $\omega_{-1} := 0$ and $\lambda_k := -B\omega_{k-1}$. These variables express the neighbor exchanges as a recursion for the graph correction.

Lemma 2. *Under the common initialization in Algorithm 1, the message rules* (9)–(10) *satisfy $\omega_{k+1} = \omega_k - \frac{\rho}{2} B^\top x_{k+1}$ and $p_k = \rho L_{\mathcal{G}} x_k - B\omega_{k-1}$. Equivalently, $p_k = \rho L_{\mathcal{G}} x_k + \lambda_k$ and $\lambda_{k+1} = \lambda_k + \frac{\rho}{2} L_{\mathcal{G}} x_k$. Moreover, $\lambda_k \in \mathrm{range}(B)$, $\mathbf{1}^\top \lambda_k = 0$ and $\mathbf{1}^\top p_k = 0$.*

Algorithm 1 SPADE-DFL at client i

Input: Local data $\mathcal{D}_i$; neighbor set $\mathcal{N}_i$; public model x_0; parameters β, ρ, τ_i, b_i; schedules $\{\eta_k, \mu_k, R_k^{\mathrm{clip}}, \sigma_{\mathrm{dp},k}\}_{k=0}^{K-1}$.

Output: Private released model $x_{i,K}$ and auxiliary states $\{z_{ij,K}\}_{j\in\mathcal{N}_i}$.

for $k = 0, \dots, K-1$ **do**
  Set $\phi_{i,k}^0 = x_{i,k}$, freeze $p_{i,k}$ by (6), and initialize the derivative-free estimator memory by (4).
  for $t = 0, \dots, \tau_i - 1$ **do**
    Sample $\mathcal{B}_{i,k}^t$, evaluate one $q_{i,h,k}^t$ per selected component using (3), and form $v_{i,k}^t$ by (5).
    Update $\phi_{i,k}^{t+1}$ by (7), replace the sampled memory pairs, and recompute $\bar{a}_{i,k}^{t+1}$.
  end for
  Set $y_{i,k+1} = \phi_{i,k}^{\tau_i}$ and $s_{i,k} = -\eta_k \sum_{t=0}^{\tau_i-1} v_{i,k}^t$.
  Clip the accumulated local update, where $\mathrm{clip}_R(0) = 0$:

$$\bar{s}_{i,k} = \mathrm{clip}_{R_k^{\mathrm{clip}}}(s_{i,k}) = s_{i,k} \min\left\{1, \frac{R_k^{\mathrm{clip}}}{\|s_{i,k}\|}\right\}.$$

  Independently sample the Gaussian release noise as

$$\nu_{i,k} \sim \mathcal{N}\left(0, \sigma_{\mathrm{dp},k}^2 I_d\right).$$

  Compute and release $x_{i,k+1}$ by (8).
  Exchange the messages in (9) and update all $z_{ij,k+1}$ by (10).
end for

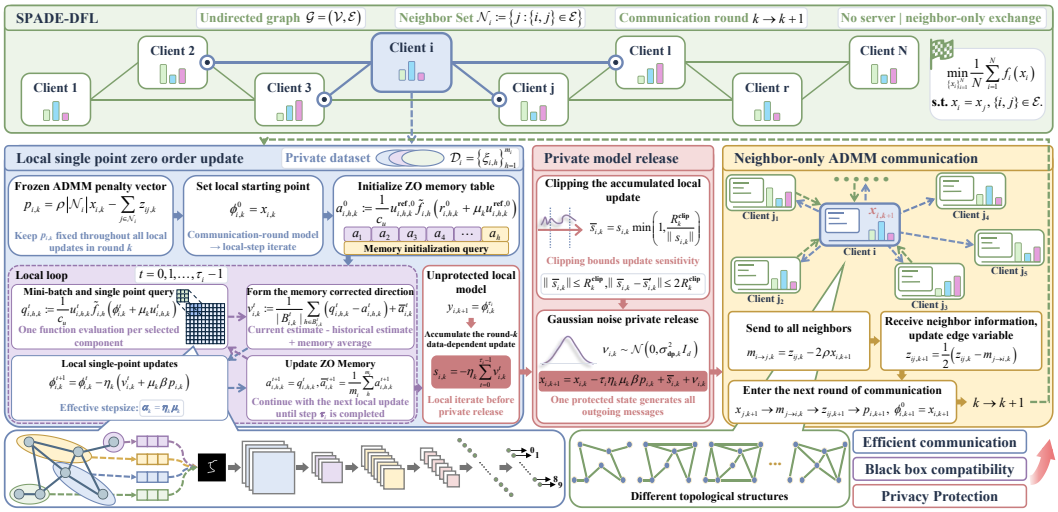

Figure 1: Workflow of SPADE-DFL: local derivative-free updates with a fixed ADMM correction, followed by a protected release and neighbor communication.

Two inputs are adjacent for client i if $\mathcal{D}_i$ is replaced by $\mathcal{D}_i'$ of the same prescribed size m_i, with all other datasets unchanged. Let $\mathcal{H}_K$ be the augmented transcript of released states and exchanged messages through round $K-1$, together with the public initialization and schedules. Given the released-state history, all directed messages and auxiliary states are deterministic. Thus, privacy of $\mathcal{H}_K$ implies privacy of any observed message transcript. For $\delta_i \in (0,1)$, define $\rho_{i,\mathrm{priv}} := \sum_{k=0}^{K-1} \frac{2(R_k^{\mathrm{clip}})^2}{\sigma_{\mathrm{dp},k}^2}$ and $\epsilon_i := \rho_{i,\mathrm{priv}} + 2\sqrt{\rho_{i,\mathrm{priv}} \log(1/\delta_i)}$. Applying adaptive composition to the Gaussian releases gives the following privacy guarantee for the full interactive transcript.

Theorem 1 (Transcript privacy)**.** *Use the public, data-independent initialization and schedules in Section 3.3. Suppose $\sigma_{\mathrm{dp},k} > 0$ for $k = 0, \dots, K-1$. Then $\mathcal{H}_K$ is $\rho_{i,\mathrm{priv}}$-zCDP and (ϵ_i, δ_i)-DP with respect to replacement of $\mathcal{D}_i$.*

Conditioning on $\mathcal{H}_k$ fixes the explicit graph correction. The clipped local increment may remain randomized, so the proof bounds the divergence between Gaussian mixtures before applying adaptive composition.

4.2 Network Stability and Disagreement

Throughout Sections 4.2–4.4, impose Assumptions 1–3 and the common initialization in Section 3.3. Use synchronized constants $\tau_i = \tau \geq 1$, $\eta_k = \eta > 0$, and $\mu_k = \mu > 0$, with $\rho, \beta > 0$ fixed within each run. The clipping and noise schedules may vary with k.

Write $\bar{\phi}_k^t := N^{-1}\sum_i \phi_{i,k}^t$ and $\bar{x}_k := N^{-1}\sum_i x_{i,k}$. Define the released-state disagreement as $\mathcal{C}_k := \frac{1}{N}\mathbb{E}[\|\Pi x_k\|^2]$. Then, the clipping and cumulative release energies are $\mathcal{E}_{\mathrm{cl},K} := \sum_{k=0}^{K-1} E_k^{\mathrm{cl}}$ and $\mathcal{E}_{\mathrm{dp},K} := \sum_{k=0}^{K-1} d\sigma_{\mathrm{dp},k}^2$, where $E_k^{\mathrm{cl}} := \frac{1}{N}\sum_{i=1}^N \mathbb{E}[\|e_{i,k}^{\mathrm{cl}}\|^2]$. Thus, Lemma 2 associates each nonzero Laplacian eigenvalue λ with the homogeneous modal matrix $M_\lambda = \left[\begin{smallmatrix} 1-a\rho\lambda & -a \\ \rho\lambda/2 & 1 \end{smallmatrix}\right]$, where $\alpha = \eta\mu$ and $a = \tau\beta\alpha$. Local updates, clipping and release noise enter as additive inputs. The homogeneous dynamics satisfy the following exact stability criterion.

Lemma 3 (Schur stability)**.** *The matrices M_λ are Schur stable for all nonzero graph modes if and only if $0 < a\rho\lambda_N < \frac{8}{3}$.*

The modal condition governs homogeneous disagreement, while the oracle and descent bounds control the stochastic inputs. The single-client case follows by removing the network terms.

4.3 Finite-Horizon Stationarity and Consensus

Define $\mathcal{S}_K := \frac{1}{K\tau}\sum_{k=0}^{K-1}\sum_{t=0}^{\tau-1}\mathbb{E}[\|\nabla F(\bar{\phi}_k^t)\|^2] + \frac{1}{K}\sum_{k=0}^{K-1}\mathcal{C}_k$ as the time-averaged stationarity–consensus criterion, and $\overline{\mathcal{E}}_{\mathrm{cl},K} := \mathcal{E}_{\mathrm{cl},K}/K$ and $\overline{\mathcal{E}}_{\mathrm{dp},K} := \mathcal{E}_{\mathrm{dp},K}/K$ as the average release energies. Combining the oracle, disagreement and descent bounds yields the following finite-horizon guarantee.

Theorem 2 (Stationarity and consensus)**.** *Suppose Assumptions 1–3 hold. Use the common initialization and the synchronized parameters in Section 4.2, with a fixed round gain $a = \tau\beta\eta\mu$ satisfying $0 < a\rho\lambda_N < 8/3$. Then, for each $K \geq 1$,*

$$\mathcal{S}_K = \mathcal{O}\left(\frac{1}{K\tau\alpha} + \frac{\eta}{\mu} + \mu^2 + \eta^2\tau^2 + \left(1 + \frac{1}{\tau\alpha^2}\right)\overline{\mathcal{E}}_{\mathrm{cl},K} + \left(1 + \frac{1}{N\tau\alpha}\right)\overline{\mathcal{E}}_{\mathrm{dp},K}\right). \tag{11}$$

The implicit constant depends only on the initial objective gap, ρ, a, the fixed graph, and the smoothness and oracle constants. It is independent of K, τ, η, μ, and the release energies.

The first four terms in (11) account for derivative-free descent and local drift; the remaining terms quantify clipping and Gaussian release. The release terms contribute to an upper bound on the time-averaged criterion $\mathcal{S}_K$, evaluated using internal iterates without additional releases.

For the non-private case, set $e_{i,k}^{\mathrm{cl}} = 0$ and $\sigma_{\mathrm{dp},k} = 0$, and write $T := K\tau$. For fixed $\eta_0, \mu_0 > 0$ and $0 < a_0 < 8/(3\rho\lambda_N)$, choose $\eta_T = \eta_0 T^{-1/2}$, $\mu_T = \mu_0 T^{-1/6}$, and $\beta_{T,\tau} = a_0/(\tau\eta_T\mu_T)$. Substituting this schedule into Theorem 2 quantifies how the local stage length affects the rate.

Corollary 1 (Non-private local-update rate)**.** *Under the conditions of Theorem 2, use the nonprivate parameter schedule with zero clipping error and release noise. If the oracle moment bound is uniform over this family of runs, then*

$$\mathcal{S}_K = \mathcal{O}\left(T^{-1/3} + \frac{\tau^2}{T}\right). \tag{12}$$

In particular, $\tau = \mathcal{O}(T^{1/3})$ preserves the $\mathcal{O}(T^{-1/3})$ order, and $\tau = \Theta(T^{1/3})$ gives $K = \Theta(T^{2/3})$.

Under this schedule, the round gain remains fixed while the individual parameters vary.

4.4 Communication and Query Complexity

Under the non-private schedule of Corollary 1, choosing $\tau = \Theta(T^{1/3})$ preserves the $\mathcal{O}(T^{-1/3})$ stationarity–consensus rate. For a tolerance $\varepsilon_{\mathrm{stat}} > 0$ on $\mathcal{S}_K$, sufficient budgets are $T = \mathcal{O}(\varepsilon_{\mathrm{stat}}^{-3})$ local updates and $K = \mathcal{O}(\varepsilon_{\mathrm{stat}}^{-2})$ communication rounds. The latter improves on the sufficient $\mathcal{O}(\varepsilon_{\mathrm{stat}}^{-3})$ round bound for the $\tau = 1$ specialization. Client i uses $m_i + \tau b_i$ function queries per round, including memory initialization, giving $\mathcal{Q}_i = Km_i + Tb_i = \mathcal{O}(m_i\varepsilon_{\mathrm{stat}}^{-2} + b_i\varepsilon_{\mathrm{stat}}^{-3})$. Each directed edge carries one d-dimensional vector per round, so the total scalar communication cost is $2|\mathcal{E}|dK = \mathcal{O}(|\mathcal{E}|d\varepsilon_{\mathrm{stat}}^{-2})$. The memory table occupies $\mathcal{O}(m_i d)$ storage in addition to the model and edge states. For private training with a constant clipping radius and uniform release-noise variance, Theorem 1 gives $\sigma_{\mathrm{dp}}^2 \propto K$ at a fixed total zCDP budget. At fixed T, increasing τ reduces both the number of releases $K = T/\tau$ and the noise variance required per release. Theorem 2 quantifies

the accompanying local-drift and release-error terms, relating these communication savings to the stationarity–consensus bound.

5 Experimental Evaluation

We evaluate SPADE-DFL on four classification tasks under heterogeneous client-data partitions. We also examine the effects of client count and privacy budget, and compare communication costs under matched local-update budgets on a bounded nonconvex problem.

5.1 Datasets and Models

We evaluate binary MNIST, binary Fashion-MNIST, binary Alzheimer's disease classification with 39 clinical features, and four-class Brain Tumor MRI. MNIST and Fashion-MNIST use 10-dimensional PCA features; Brain Tumor MRI uses frozen ResNet-18 V1 features reduced to 10 dimensions by PCA fitted on the training split, followed by a 40-parameter linear softmax classifier. Comparisons cover IID, Dirichlet ($\alpha = 0.6, 0.3$), and pathological client partitions, sharing partitions and random seeds across methods. Appendix F provides complete setup details and dataset links.

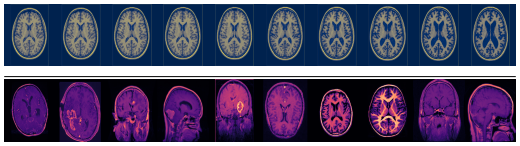

Figure 2: Illustrative brain images related to Alzheimer's disease and brain tumors.

5.2 Comparative Evaluation under Heterogeneous Client Data

We compare SPADE-DFL with LT-ADMM-VR, LT-ADMM-DP, DP-SGD, DP-FedAvg, 1P-DSG, and 1P-DSGT. The LT-ADMM-VR, LT-ADMM-DP, DP-SGD, and DP-FedAvg implementations use symmetric two-point component-loss estimates, while 1P-DSG, 1P-DSGT, and SPADE-DFL use one-point estimates. Function-query budgets and privacy settings vary across methods.

The comparison uses 50 rounds and 30 paired random seeds. The decentralized implementations use $N = 31$ clients on an undirected Erdős–Rényi graph with edge probability 0.3. The centralized DP-SGD baseline uses pooled data with $N = 1$, so its results are shared across the client-data partitions. SPADE-DFL uses $\tau = 2$ local updates for MNIST, Fashion-MNIST, and Brain Tumor MRI, and $\tau = 8$ for Alzheimer's disease. For the comparative experiments, SPADE-DFL reinitializes the estimator memory at the beginning of each round using (4) and all settings remain consistent within the same group of tests.

For MNIST, Fashion-MNIST, and Brain Tumor MRI, the curves and final-round values are summarized over 30 random seeds using the arithmetic mean and population standard deviation. For Alzheimer's disease, the same statistics are calculated over 150 fold–seed trajectories obtained from five stratified folds and 30 seeds per fold. All accuracy values are reported as percentages. Accuracy trajectories over 50 communication rounds are plotted in Figs. 3–6. From left to right, the panels correspond to IID, Dirichlet $\alpha = 0.6$, Dirichlet $\alpha = 0.3$, and pathological partitions. The curves show the arithmetic mean, and the shaded regions show $\pm$ one population standard deviation. The corresponding final-round accuracies are summarized in Table 1. Boldface marks the largest mean in each row, underlining marks the largest competing mean, and Δ_{best} denotes their difference in percentage points.

Table 1: Test accuracy at round 50 under the configurations in Section 5.2.

Dataset	Distribution	LT-ADMM-VR	LT-ADMM-DP	DP-SGD	DP-FedAvg	1P-DSG	1P-DSGT	SPADE-DFL	Δ_{best}
MNIST	IID	85.30±7.35	92.22±3.97	55.81±13.16	71.69±8.96	62.15±12.91	76.58±11.20	**95.40±3.24**	+3.18
	Dir. ($\alpha = 0.6$)	83.95±7.29	92.04±5.01	55.81±13.16	71.68±9.11	61.81±12.81	75.68±11.33	**95.31±2.68**	+3.27
	Dir. ($\alpha = 0.3$)	85.07±6.97	91.55±4.75	55.81±13.16	71.75±9.15	61.15±12.78	75.45±11.17	**95.15±3.13**	+3.60
	Pathological	85.21±7.48	92.05±4.85	55.81±13.16	71.69±9.11	61.98±12.97	75.97±11.81	**95.37±3.08**	+3.32
Fashion-MNIST	IID	84.95±5.94	88.74±4.03	57.33±13.79	70.67±9.98	58.01±12.37	74.43±11.36	**93.28±2.21**	+4.54
	Dir. ($\alpha = 0.6$)	84.48±6.51	88.42±4.68	57.33±13.79	70.89±10.04	57.90±13.03	74.92±10.46	**93.25±1.87**	+4.83
	Dir. ($\alpha = 0.3$)	84.12±6.86	89.13±3.22	57.33±13.79	70.77±10.18	58.03±12.94	75.09±10.63	**93.36±2.18**	+4.23
	Pathological	84.39±6.13	89.32±4.29	57.33±13.79	70.49±10.30	57.69±12.61	75.11±10.99	**93.17±2.22**	+3.85
Alzheimer's disease	IID	71.28±3.18	73.99±2.64	68.47±3.44	71.10±3.10	57.11±4.41	64.86±4.08	**77.27±2.91**	+3.28
	Dir. ($\alpha = 0.6$)	70.34±3.32	73.10±3.24	68.47±3.44	69.56±3.43	56.20±4.95	63.56±4.49	**77.31±2.86**	+4.21
	Dir. ($\alpha = 0.3$)	69.33±3.93	72.02±3.60	68.47±3.44	68.18±3.51	56.49±4.75	63.27±4.54	**75.73±3.62**	+3.71
	Pathological	68.56±3.72	71.61±3.02	68.47±3.44	70.17±3.05	55.42±4.93	61.84±4.95	**75.64±2.58**	+4.03
Brain Tumor MRI	IID	51.46±5.42	60.06±4.47	61.40±3.79	73.52±0.75	32.28±6.81	47.35±7.86	**74.38±0.52**	+0.86
	Dir. ($\alpha = 0.6$)	52.09±7.11	60.81±4.06	61.40±3.79	73.28±0.79	32.23±7.24	47.95±7.35	**74.35±0.55**	+1.07
	Dir. ($\alpha = 0.3$)	50.50±7.08	58.01±5.35	61.40±3.79	72.96±0.92	32.81±6.40	48.22±6.26	**74.07±0.80**	+1.11
	Pathological	53.81±6.08	59.61±4.86	61.40±3.79	73.20±0.87	32.03±7.24	48.06±7.70	**74.44±0.63**	+1.24

SPADE-DFL achieved the highest mean test accuracy at round 50 in all 16 dataset–partition combinations reported in Table 1. The gains over the best competing method ranged from 0.86 percentage

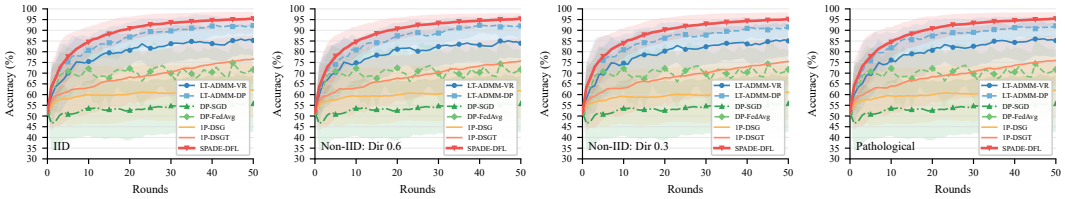

Figure 3: Test accuracy on binary MNIST.

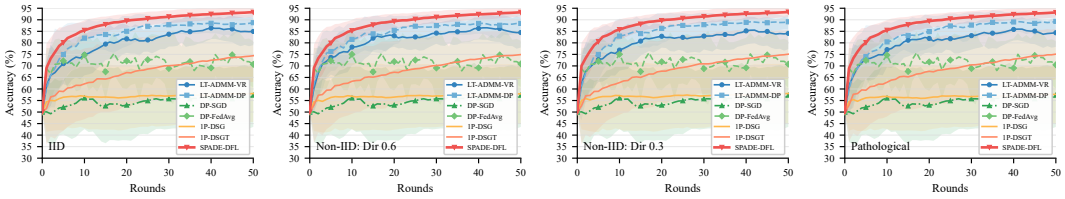

Figure 4: Test accuracy on binary Fashion-MNIST.

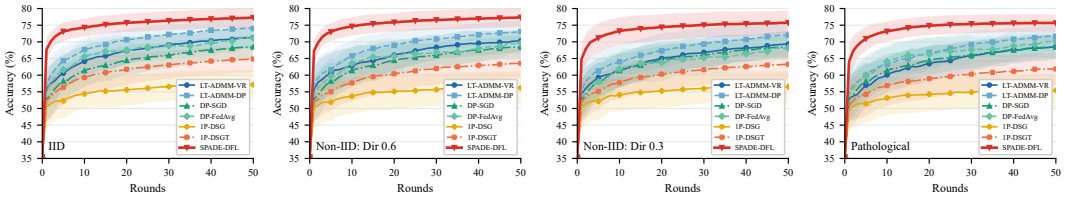

Figure 5: Test accuracy on Alzheimer's disease.

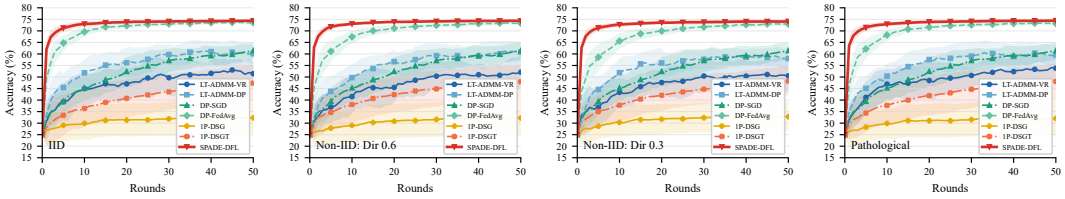

Figure 6: Test accuracy on Brain Tumor MRI.

points on IID Brain Tumor MRI to 4.83 percentage points on Fashion-MNIST with Dirichlet $\alpha = 0.6$. Across the four partitions, SPADE-DFL achieved mean accuracies of 95.15–95.40% on MNIST, 93.17–93.36% on Fashion-MNIST, 75.64–77.31% on Alzheimer's disease, and 74.07–74.44% on Brain Tumor MRI.

5.3 Sensitivity to Client Count and Privacy Budget

Table 2: Validation accuracy under varying client counts and privacy budgets.

Dataset	Client count ($\varepsilon = 32$)					Privacy budget ($N = 100$)				
	$N = 30$	$N = 40$	$N = 50$	$N = 75$	$N = 100$	$\varepsilon = 4$	$\varepsilon = 8$	$\varepsilon = 16$	$\varepsilon = 24$	$\varepsilon = 32$
MNIST	99.21±0.15	99.03±0.15	99.21±0.19	99.21±0.23	99.15±0.11	93.32±8.71	97.98±1.79	99.05±0.12	99.15±0.16	99.15±0.11
Fashion-MNIST	94.22±0.67	92.28±0.98	92.23±1.03	93.00±0.90	93.55±0.70	92.50±2.18	93.38±1.81	93.43±1.01	93.58±0.88	93.55±0.70
Alzheimer's disease	72.16±0.79	72.16±1.43	72.84±0.89	72.49±1.06	73.00±0.48	61.05±3.17	67.26±2.55	71.56±0.74	72.81±0.76	73.00±0.48
Brain Tumor MRI	65.46±3.72	68.82±3.56	65.00±2.34	70.46±3.02	70.96±3.40	43.29±9.92	54.00±6.73	64.18±5.50	69.25±4.60	70.96±3.40

We evaluate SPADE-DFL's validation accuracy under Dirichlet partitions with $\alpha = 0.3$, as reported in Table 2. MNIST, Fashion-MNIST, and Alzheimer's disease use balanced-capacity partitions, whereas Brain Tumor MRI uses the ordinary Dirichlet partition recorded in the adopted runs. The reported privacy budgets use client-level replacement adjacency with $\delta = 10^{-5}$, where adjacent inputs replace one client's entire fixed-size dataset while leaving all other clients' fixed preprocessed datasets unchanged; the accountant composes one Gaussian release per client per round with sensitivity $2R_k^{\text{clip}}$, and the guarantee applies to the resulting client datasets conditional on frozen preprocessing and partitioning.

Accuracy varies modestly across the tested client counts, with Brain Tumor MRI varying most. Relaxing privacy from $\varepsilon = 4$ to 32 gives the largest gain on Brain Tumor MRI (43.29% to 70.96%). MNIST changes little beyond $\varepsilon = 16$, whereas Fashion-MNIST varies by about one percentage point (Appendix G.3).

5.4 Client Heterogeneity and Roundwise Comparisons

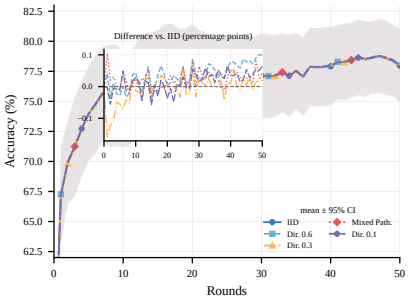

Figure 7: Test accuracy across five MNIST partitions.

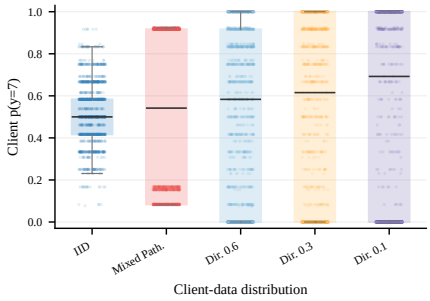

Figure 8: Client digit-7 proportions by partition.

A separate heterogeneity experiment evaluates SPADE-DFL on binary MNIST using $N = 1000$ clients on a degree-two ring, $\tau = 8$, and $(\varepsilon, \delta) = (8, 10^{-5})$. The experiment uses 50 communication rounds and 30 paired random seeds. The accuracy curves represent arithmetic means, with shaded regions indicating two-sided 95% Student's t confidence intervals, as shown in Fig. 7. The inset reports the matched-seed mean accuracy difference relative to the IID partition.

Client-level digit-7 proportions are evaluated under IID, pathological (Mixed Path), and Dirichlet partitions with $\alpha \in \{0.6, 0.3, 0.1\}$, as depicted in Fig. 8. The mean accuracy curves remain closely aligned across the five partitions, as illustrated in Fig. 7. Greater variation in digit-7 proportions across clients is observed under non-IID partitions, as shown in Fig. 8. Each point in the roundwise analysis represents one method–partition–round comparison relative to LT-ADMM-VR, as displayed in Fig. 9. The horizontal coordinate is the paired mean accuracy difference over 30 seeds. We use two-sided paired Wilcoxon signed-rank tests, with Benjamini–Hochberg correction across the 1,200 comparisons. These tests provide comparisons along the training trajectories, whose successive rounds are statistically dependent.

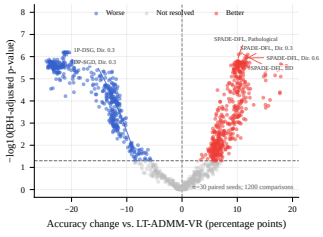

Figure 9: Accuracy differences relative to LT-ADMM-VR.

5.5 Communication–Optimization Trade-off

We compare communication after every local update ($\tau = 1$) with the prescribed local-update schedule $\tau = T^{1/3}$, as presented in Fig. 10. The experiment uses $N = 20$ clients arranged in a cycle, model dimension $d = 20$, and total local-update budgets $T \in \{1000, 3375, 10648, 29791\}$. These budgets correspond to $\tau \in \{10, 15, 22, 31\}$ under the local-update schedule.

Cumulative communication is measured as $2|\mathcal{E}|dK$ transmitted model scalars, where $K = T/\tau$. The local-update schedule reduced communication by factors of 10, 15, 22, and 31 for the four update budgets. At $T = 29{,}791$, the communication volume was 768,800 scalars with $\tau = 31$ and 23,832,800 scalars with $\tau = 1$, a reduction of 96.8%.

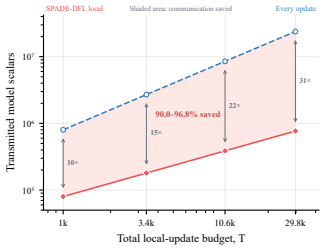

Figure 10: Communication cost under matched local-update budgets.

The empirical stationarity–consensus criterion decreased with the update budget under both schedules. At $T = 29{,}791$, the 96.8% communication saving came with a larger mean residual over 30 paired seeds: 0.010392 for $\tau = 31$ versus 0.000372 for $\tau = 1$.

6 Conclusion

SPADE-DFL reduces neighbor communication through local function evaluations while preserving the nonprivate convergence order. The analysis characterizes how local computation drives network disagreement, establishing that the communication interval can grow with the training budget and yield sublinear communication under the stated assumptions. For private training, isolating the accumulated data-dependent increment from graph feedback permits client-level protection of the full interactive transcript through one release per round and makes the optimization cost of privacy explicit. Experiments across four classification tasks show higher mean test accuracy under the tested configurations. Future work will extend these guarantees to time-varying networks and asynchronous client updates.

AI USE STATEMENT

Generative AI tools were used solely for language polishing, including grammar, wording, and readability. The authors take full responsibility for the final content of this manuscript.

REFERENCES

Sulaiman A. Alghunaim. Local exact-diffusion for decentralized optimization and learning. *IEEE Transactions on Automatic Control*, 69(11):7371–7386, 2024. doi: 10.1109/TAC.2024.3383271.

Youssef Allouah, Anastasia Koloskova, Aymane El Firdoussi, Martin Jaggi, and Rachid Guerraoui. The privacy power of correlated noise in decentralized learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 1115–1143. PMLR, 2024.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011. doi: 10.1561/2200000016.

Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pp. 635–658. Springer, 2016.

Edwige Cyffers, Mathieu Even, Aurélien Bellet, and Laurent Massoulié. Muffliato: Peer-to-peer privacy amplification for decentralized optimization and averaging. In *Advances in Neural Information Processing Systems*, volume 35, pp. 15889–15902, 2022. doi: 10.52202/068431-1156.

Edwige Cyffers, Aurélien Bellet, and Debabrota Basu. From noisy fixed-point iterations to private ADMM for centralized and federated learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 6683–6711. PMLR, 2023.

Lisang Ding, Kexin Jin, Bicheng Ying, Kun Yuan, and Wotao Yin. DSGD-CECA: Decentralized SGD with communication-optimal exact consensus algorithm. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8067–8089. PMLR, 2023.

Wenzhi Fang, Ziyi Yu, Yuning Jiang, Yuanming Shi, Colin N. Jones, and Yong Zhou. Communication-efficient stochastic zeroth-order optimization for federated learning. *IEEE Transactions on Signal Processing*, 70:5058–5073, 2022. doi: 10.1109/TSP.2022.3214122.

Tanmay Gautam, Youngsuk Park, Hao Zhou, Parameswaran Raman, and Wooseok Ha. Variance-reduced zeroth-order methods for fine-tuning language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15180–15208. PMLR, 2024.

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013. doi: 10.1137/120880811.

Xuchen Gong and Tian Li. Private zeroth-order optimization with public data. In *Advances in Neural Information Processing Systems*, volume 38, pp. 58619–58665, 2025. doi: 10.52202/085713-1963.

Pengxin Guo, Runxi Wang, Shuang Zeng, Jinjing Zhu, Haoning Jiang, Yanran Wang, Yuyin Zhou, Feifei Wang, Hui Xiong, and Liangqiong Qu. Exploring the vulnerabilities of federated learning: A deep dive into gradient inversion attacks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(4):4810–4826, 2026. doi: 10.1109/TPAMI.2025.3646639.

Yutong He, Xinmeng Huang, and Kun Yuan. Unbiased compression saves communication in distributed optimization: When and how much? In *Advances in Neural Information Processing Systems*, volume 36, pp. 47991–48020, 2023. doi: 10.52202/075280-2081.

Youqing Hua, Shuai Liu, Yiguang Hong, and Wei Ren. Distributed stochastic zeroth-order optimization with compressed communication. *IEEE Transactions on Automatic Control*, 71(2): 1294–1301, 2026. doi: 10.1109/TAC.2025.3610109.

Feihu Huang, Shangqian Gao, Jian Pei, and Heng Huang. Nonconvex zeroth-order stochastic ADMM methods with lower function query complexity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–13, 2024. doi: 10.1109/TPAMI.2023.3347082. Early Access.

Rustem Islamov, Yuan Gao, and Sebastian U. Stich. Towards faster decentralized stochastic optimization with communication compression. In *Proceedings of the 13th International Conference on Learning Representations*, 2025.

Peter Kairouz, H. Brendan McMahan, Brendan Avent, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. doi: 10.1561/2200000083.

Anastasia Koloskova, Hadrien Hendrikx, and Sebastian U. Stich. Revisiting gradient clipping: Stochastic bias and tight convergence guarantees. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17343–17363. PMLR, 2023.

Jueyou Li, Chaojie Li, Jing Fan, and Tingwen Huang. Online distributed stochastic gradient algorithm for nonconvex optimization with compressed communication. *IEEE Transactions on Automatic Control*, 69(2):936–951, 2024a. doi: 10.1109/TAC.2023.3327183.

Qiongxiu Li, Jaron Skovsted Gundersen, Milan Lopuhaä-Zwakenberg, and Richard Heusdens. Adaptive differentially quantized subspace perturbation (ADQSP): A unified framework for privacy-preserving distributed average consensus. *IEEE Transactions on Information Forensics and Security*, 19:1780–1793, 2024b. doi: 10.1109/TIFS.2023.3343599.

Yichuan Li, Petros G. Voulgaris, Dušan M. Stipanović, and Nikolaos M. Freris. Communication efficient curvature aided primal-dual algorithms for decentralized optimization. *IEEE Transactions on Automatic Control*, 68(11):6573–6588, 2023. doi: 10.1109/TAC.2023.3244904.

Zhe Li, Bicheng Ying, Zidong Liu, Chaosheng Dong, and Haibo Yang. Achieving dimension-free communication in federated learning via zeroth-order optimization. In *Proceedings of the 13th International Conference on Learning Representations*, 2025.

Qing Ling, Wei Shi, Gang Wu, and Alejandro Ribeiro. DLM: Decentralized linearized alternating direction method of multipliers. *IEEE Transactions on Signal Processing*, 63(15):4051–4064, 2015. doi: 10.1109/TSP.2015.2436358.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D. Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 53038–53075, 2023. doi: 10.52202/075280-2308.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017.

Elissa Mhanna and Mohamad Assaad. Single point-based distributed zeroth-order optimization with a non-convex stochastic objective function. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 24701–24719. PMLR, 2023.

Elissa Mhanna and Mohamad Assaad. Zero-order one-point gradient estimate in consensus-based distributed stochastic optimization. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=tkswb7XoYB.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium*, pp. 263–275, 2017. doi: 10.1109/CSF.2017.11.

Konstantin Mishchenko, Grigory Malinovsky, Sebastian Stich, and Peter Richtárik. ProxSkip: Yes! Local gradient steps provably lead to communication acceleration! Finally! In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 15750–15769. PMLR, 2022.

Roula Nassif, Stefan Vlaski, Marco Carpentiero, Vincenzo Matta, and Ali H. Sayed. Differential error feedback for communication-efficient decentralized learning. *IEEE Transactions on Signal Processing*, 73:1905–1921, 2025. doi: 10.1109/TSP.2025.3564416.

Xiaoxing Ren, Nicola Bastianello, Karl H. Johansson, and Thomas Parisini. Communication-efficient stochastic distributed learning. *IEEE Transactions on Automatic Control*, 71(9):5741–5756, 2026a. doi: 10.1109/TAC.2026.3675513.

Xiaoxing Ren, Yuwen Ma, Nicola Bastianello, Karl H. Johansson, Thomas Parisini, and Andreas A. Malikopoulos. Communication-efficient distributed learning with differential privacy. *arXiv preprint arXiv:2604.02558*, 2026b. doi: 10.48550/arXiv.2604.02558.

Elsa Rizk, Stefan Vlaski, and Ali H. Sayed. Enforcing privacy in distributed learning with performance guarantees. *IEEE Transactions on Signal Processing*, 71:3385–3398, 2023. doi: 10.1109/TSP.2023.3316590.

Wei Shi, Qing Ling, Kun Yuan, Gang Wu, and Wotao Yin. On the linear convergence of the ADMM in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 62(7):1750–1761, 2014. doi: 10.1109/TSP.2014.2304432.

Yifan Shi, Kang Wei, Li Shen, Yingqi Liu, Xueqian Wang, Bo Yuan, and Dacheng Tao. Toward the flatter landscape and better generalization in federated learning under client-level differential privacy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(12):11632–11643, 2025. doi: 10.1109/TPAMI.2025.3597922.

Zhuoqing Song, Lei Shi, Shi Pu, and Ming Yan. Compressed gradient tracking for decentralized optimization over general directed networks. *IEEE Transactions on Signal Processing*, 70:1775–1787, 2022. doi: 10.1109/TSP.2022.3160238.

Yan Sun, Li Shen, and Dacheng Tao. Toward understanding generalization and stability gaps between centralized and decentralized federated learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(4):4744–4755, 2026. doi: 10.1109/TPAMI.2025.3647762.

Luqing Wang, Shaofu Yang, Yifan Wan, Wenying Xu, and Min-Ling Zhang. Privacy preserving decentralized learning with positive-incentive noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(7):8520–8534, 2026. doi: 10.1109/TPAMI.2026.3672569.

Tongle Wu, Zhize Li, and Ying Sun. The effectiveness of local updates for decentralized learning under data heterogeneity. *IEEE Transactions on Signal Processing*, 73:751–765, 2025. doi: 10.1109/TSP.2025.3533208.

Lei Xu, Xinlei Yi, Chao Deng, Yang Shi, Tianyou Chai, and Tao Yang. Quantized zeroth-order gradient tracking algorithm for distributed nonconvex optimization under Polyak–Łojasiewicz condition. *IEEE Transactions on Cybernetics*, 54(10):5746–5758, 2024. doi: 10.1109/TCYB.2024.3384924.

Haishan Ye, Zhichao Huang, Cong Fang, Chris Junchi Li, and Tong Zhang. Hessian-aware zeroth-order optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(6):4869–4877, 2025. doi: 10.1109/TPAMI.2025.3548810.

Runze You and Shi Pu. B-ary tree push-pull method is provably efficient for distributed learning on heterogeneous data. In *Advances in Neural Information Processing Systems*, volume 37, pp. 97523–97561, 2024. doi: 10.52202/079017-3094.

Kun Yuan, Sulaiman A. Alghunaim, and Xinmeng Huang. Removing data heterogeneity influence enhances network topology dependence of decentralized SGD. *Journal of Machine Learning Research*, 24(280):1–53, 2023.

Shahryar Zehtabi, Dong-Jun Han, Rohit Parasnis, Seyyedali Hosseinalipour, and Christopher G. Brinton. Decentralized sporadic federated learning: A unified algorithmic framework with convergence guarantees. In *Proceedings of the 13th International Conference on Learning Representations*, 2025.

Zhiyuan Zhai, Xiaojun Yuan, Xin Wang, and Geoffrey Ye Li. Decentralized federated learning with distributed aggregation weight optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3):3899–3910, 2026. doi: 10.1109/TPAMI.2025.3640709.

Liang Zhang, Bingcong Li, Kiran Koshy Thekumparampil, Sewoong Oh, and Niao He. DPZero: Private fine-tuning of language models without backpropagation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 59210–59246. PMLR, 2024.

Haoyu Zhao, Boyue Li, Zhize Li, Peter Richtárik, and Yuejie Chi. BEER: Fast $O(1/T)$ rate for decentralized nonconvex optimization with communication compression. In *Advances in Neural Information Processing Systems*, volume 35, pp. 31653–31667, 2022. doi: 10.52202/068431-2295.

APPENDIX CONTENTS

A Complete Related Work . . . 15

Table 3 Selected algorithmic features of the methods discussed in Related Work . . . 17

B Complete Single-Point Estimation and Memory Properties . . . 18

Proof of Lemma 1 . . . 19

C Message Recursion and Transcript Privacy . . . 20

Proof of Lemma 2 . . . 20

Proof of Theorem 1 . . . 20

D Schur Stability and Disagreement Bounds . . . 20

Full statement of Lemma 3 . . . 21

Proof of Lemma 3 . . . 21

Remark 1: Single-client case . . . 22

E Descent and Convergence Bounds . . . 22

Lemma 4: Within-round disagreement . . . 22

Proof of Lemma 4 . . . 22

Lemma 5: One-round descent . . . 22

Proof of Lemma 5 . . . 22

Proof of Theorem 2 . . . 23

Proof of Corollary 1 . . . 24

Uniform constants and transcript accounting . . . 24

Remark 2: Convex objectives . . . 24

F Datasets, Models, and Data Sources . . . 25

G Additional Method and Experimental Details . . . 25

Extended method overview and contributions . . . 25

Equivalent updates and release operations . . . 26

Sensitivity to client count and privacy budget . . . 27

A Complete Related Work

A.1 Communication in Decentralized Learning

The benefit of local computation depends on whether progress made between exchanges survives the disagreement generated by heterogeneous objectives. Analyses of local gradient descent and gradient tracking make this dependence explicit (Wu et al., 2025). ProxSkip establishes communication acceleration through randomized synchronization for strongly convex problems (Mishchenko et al., 2022). Local exact diffusion incorporates bias correction into local training (Alghunaim, 2024), while local training ADMM holds a neighbor correction fixed over several stochastic updates (Ren et al., 2026a). A differentially private variant applies clipping and Gaussian perturbation to stochastic gradients at each local step while retaining one neighbor-exchange phase per round (Ren et al., 2026b). Its analysis establishes privacy under single-record addition or removal and a stationarity bound for nonconvex objectives. The communication cost also depends on the mixing protocol. DSGD with CECA uses an exact consensus schedule (Ding et al., 2023), whereas tree based push pull limits the number of active neighbors (You & Pu, 2024). Related analyses show how heterogeneity correction improves the dependence of transient behavior on topology (Yuan et al., 2023). Work on DFL examines this relationship through aggregation weight optimization (Zhai et al., 2026) and the stability of decentralized training (Sun et al., 2026). Compression addresses the amount transmitted at each exchange, with its overall benefit depending on the convergence cost of smaller messages (He et al., 2023). Compressed gradient tracking treats communication over directed networks (Song et al., 2022). Differential error feedback reuses compression residuals (Nassif et al., 2025), while BEER develops compression with gradient tracking for nonconvex objectives (Zhao et al., 2022). MoTEF uses momentum tracking to control stochastic error within an error feedback scheme (Islamov et al., 2025). For objectives that vary over time, compressed distributed methods connect communication to online regret (Li et al., 2024a).

A.2 Learning from Function Values

Zeroth-order optimization studies how function evaluations can provide enough directional information for learning (Ghadimi & Lan, 2013). Single-point distributed methods use one noisy evaluation per update (Mhanna & Assaad, 2023; 2024). The accuracy of the inferred direction can also be improved through curvature information (Ye et al., 2025) or stochastic ADMM constructions with explicit query complexity (Huang et al., 2024). In networked problems, compressed stochastic methods incorporate transmission error into the analysis (Hua et al., 2026). Quantized gradient tracking with deterministic zeroth-order estimates yields linear convergence under the Polyak Łojasiewicz condition (Xu et al., 2024). Function evaluations also change the implementation cost of local learning. FedZO performs several local updates between server aggregations (Fang et al., 2022), while DeComFL represents communicated information by scalars to remove the dependence of the payload on model dimension (Li et al., 2025). For language model adaptation, MeZO avoids backpropagation by estimating directions from paired forward evaluations (Malladi et al., 2023). Its SVRG extension uses reference information to improve the optimization process (Gautam et al., 2024). These developments connect the choice of estimator to the work performed by the local solver. For decentralized solvers, ADMM expresses coordination through equality constraints on neighboring models (Boyd et al., 2011; Shi et al., 2014). Linearization replaces the local primal solve with an explicit approximation (Ling et al., 2015). Local training allows this computation to continue between exchanges (Ren et al., 2026a). Curvature aided methods use gradient directions or Newton approximations within local primal and dual updates without inner communication loops for convex composite objectives (Li et al., 2023).

A.3 Privacy across Distributed Interactions

Privacy in decentralized learning depends on what an observer can infer from a history of exchanges. Graph homomorphic noise connects the protection of communicated states to learning performance (Rizk et al., 2023). Adaptive differentially quantized subspace perturbation exploits the consensus structure to address privacy with compressed communication (Li et al., 2024b). Positive incentive noise designs use graph interactions to regulate the resulting utility (Wang et al., 2026). Muffliato analyzes privacy amplification under a pairwise network observation model (Cyffers et al., 2022). DECOR instead constructs Gaussian perturbations that cancel across neighboring clients us-

ing shared secret randomness (Allouah et al., 2024). Private ADMM connects iterative optimization to a noisy fixed point formulation (Cyffers et al., 2023). Concentrated privacy and Rényi privacy provide composition rules for successive releases (Bun & Steinke, 2016; Mironov, 2017). The effect of a release also depends on how the local update is prepared. Clipping can introduce persistent stochastic bias (Koloskova et al., 2023). DP FedSAM examines private local training through sharpness aware optimization with update sparsification (Shi et al., 2025). For learning from function values, DPZero develops private model adaptation without backpropagation (Zhang et al., 2024). PAZO uses public data to guide the private gradient approximation under a similarity assumption between the two data sources (Gong & Li, 2025).

SPADE-DFL quantifies the communication savings attainable from local single-point function evaluations. Under uniform query-moment bounds, the prescribed nonprivate schedule permits the local training length to grow with the computation budget while preserving the stationarity and consensus convergence order. For private training, the accumulated local update is clipped and perturbed once per round, and client-level privacy is established for the full interactive transcript under replacement of an entire fixed-size local dataset. These distinctions from the methods discussed above are summarized in Appendix Table 3.

Table 3: Selected algorithmic features of the methods discussed in Related Work.

Method	Communication reduction	Local updates	Single-point oracle	Component memory	ADMM updates	Differential privacy
Local training (Wu et al., 2025; Alghunaim, 2024; Mishchenko et al., 2022; Fang et al., 2022; Li et al., 2025)	✓	✓				
LT-ADMM (Ren et al., 2026a)	✓	✓			✓	
LT-ADMM-VR (Ren et al., 2026a)	✓	✓		✓	✓	
LT-ADMM-DP (Ren et al., 2026b)	✓	✓			✓	✓
Communication-saving methods (Ding et al., 2023; You & Pu, 2024; Song et al., 2022; Nassif et al., 2025; Zhao et al., 2022; Islamov et al., 2025; Li et al., 2024a; Hua et al., 2026; Xu et al., 2024)	✓					
Nonprivate optimization (Yuan et al., 2023; Zhai et al., 2026; Ye et al., 2025; Malladi et al., 2023; Gautam et al., 2024)						
Single-point methods (Mhanna & Assaad, 2023; 2024)			✓			
ADMM methods (Huang et al., 2024; Shi et al., 2014; Ling et al., 2015)					✓	
ADMM variants (Li et al., 2023; 2024b)	✓				✓	
Private learning (Rizk et al., 2023; Allouah et al., 2024; Wang et al., 2026; Cyffers et al., 2022; Zhang et al., 2024; Gong & Li, 2025)						✓
Private ADMM (Cyffers et al., 2023)					✓	✓
DP-FedSAM (top k) (Shi et al., 2025)	✓	✓				✓
SPADE-DFL	✓	✓	✓	✓	✓	✓

B Complete Single-Point Estimation and Memory Properties

Each sampled component provides one noisy function value from which a derivative-free direction can be formed. Let $\mathcal{F}$ denote the information available before a query at an $\mathcal{F}$-measurable point x. Client i observes $\widetilde{f}_{i,h}(x+\mu u) = f_{i,h}(x+\mu u)+\zeta$, where $\mu > 0$ is the smoothing radius, $u \in \mathbb{R}^d$ is the current query direction and ζ is the query noise, respectively. The following assumption adapts the one-point perturbation model to the generated query points.

Assumption 3 ((Mhanna & Assaad, 2023; 2024)). *For each generated query at an $\mathcal{F}$-measurable point, there exist positive constants $c_u, R_u, M_2 > 0$ such that $\mathbb{E}[u \mid \mathcal{F}] = 0$, $\mathbb{E}\big[uu^\top \mid \mathcal{F}\big] = c_u I_d$ and $\|u\| \le R_u$ almost surely. Moreover, $\mathbb{E}[\zeta \mid \mathcal{F}, u] = 0$ and $\mathbb{E}\Big[|\widetilde{f}_{i,h}(x+\mu u)|^2 \mid \mathcal{F}\Big] \le M_2$.*

The query-moment condition admits smooth objectives on an unconstrained parameter domain. If $|f_{i,h}(z)| \le B_f$ for every i, h and $z \in \mathbb{R}^d$, and $\mathbb{E}[\zeta^2 \mid \mathcal{F}, u] \le \sigma_\zeta^2$, conditional centering of the query noise gives $\mathbb{E}\Big[|\widetilde{f}_{i,h}(x+\mu u)|^2 \mid \mathcal{F}\Big] \le B_f^2 + \sigma_\zeta^2$. A concrete example is the squared probability loss $f(x) = \frac{1}{2}\|\operatorname{softmax}(Wa) - y\|^2$, where $x = \operatorname{vec}(W)$, the feature vector a has uniformly bounded norm, and y is a one-hot label. This loss is globally smooth and takes values in $[0,1]$ throughout the parameter space. Thus, bounded smooth losses with uniformly bounded conditional noise variance provide a query-moment bound independent of the query locations and smoothing radii. Taking B_f and σ_ζ independent of the training budget also provides the uniformity required by the convergence rates. Under Assumption 3, the local single-point estimator is

$$q_{i,h}(x; u, \zeta) = \frac{1}{c_u} u \widetilde{f}_{i,h}(x+\mu u). \tag{3}$$

Each evaluation of (3) requires one function value. The estimator is used without division by μ. At local step t of round k, client i evaluates (3) at $x = \phi_{i,k}^t$ with smoothing radius μ_k and direction $u_{i,h,k}^t$; the resulting query is denoted by $q_{i,h,k}^t$.

In round k, client i starts from the released model $x_{i,k}$ and performs τ_i local updates with mini-batch size $1 \le b_i \le m_i$. Set $\phi_{i,k}^0 = x_{i,k}$. To reuse component-level single-point information, one memory entry is initialized for every $h \in \{1, \dots, m_i\}$ by setting $r_{i,h,k}^0 = x_{i,k}$, drawing a reference direction $u_{i,h,k}^{\mathrm{ref},0}$, and storing

$$a_{i,h,k}^0 = \frac{1}{c_u} u_{i,h,k}^{\mathrm{ref},0} \widetilde{f}_{i,h}\left(r_{i,h,k}^0 + \mu_k u_{i,h,k}^{\mathrm{ref},0}\right). \tag{4}$$

The initial table average is $\bar{a}_{i,k}^0 := m_i^{-1} \sum_{h=1}^{m_i} a_{i,h,k}^0$, and this initialization uses one function query per component at the beginning of the round.

Let $\mathcal{F}_{k,t}$ contain the public transcript, released and auxiliary states, frozen penalties, current local iterates, memory table, and all randomness revealed before local step (k,t). The current mini-batches, query directions, and query noises are excluded. Conditional on $\mathcal{F}_{k,t}$, each $\mathcal{B}_{i,k}^t$ is uniform over the subsets of size b_i and is independent of the current query randomization; current batches are independent across clients. With $\bar{a}_{i,k}^t := m_i^{-1} \sum_{h=1}^{m_i} a_{i,h,k}^t$, define

$$v_{i,k}^t = \frac{1}{|\mathcal{B}_{i,k}^t|} \sum_{h \in \mathcal{B}_{i,k}^t} \left(q_{i,h,k}^t - a_{i,h,k}^t\right) + \bar{a}_{i,k}^t. \tag{5}$$

After forming $v_{i,k}^t$, each sampled pair is replaced by $(r_{i,h,k}^{t+1}, a_{i,h,k}^{t+1}) := (\phi_{i,k}^t, q_{i,h,k}^t)$, while unsampled entries remain unchanged and $\bar{a}_{i,k}^{t+1}$ is recomputed.

To state the oracle and memory errors, define

$$r_{i,h}(x,\mu) := \mathbb{E}[q_{i,h}(x; u, \zeta) \mid \mathcal{F}] - \mu \nabla f_{i,h}(x),$$
$$r_{i,k}^t := \mathbb{E}\big[v_{i,k}^t \mid \mathcal{F}_{k,t}\big] - \mu_k \nabla f_i(\phi_{i,k}^t).$$

For compactness, set $c_r := \frac{L_f R_u^3}{2c_u}$ and $Q^2 := \frac{R_u^2 M_2}{c_u^2}$. The two statements below characterize the mean and moment properties of the unnormalized single-query estimator under Assumption 3 and those of the memory recursion in (4)–(5), respectively.

Lemma 1. *Under Assumptions 1 and 3, the following statements hold.*

(i) For each $\mathcal{F}$-measurable query point x, it holds that $\mathbb{E}[q_{i,h}(x;u,\zeta) \mid \mathcal{F}] = \mu\nabla f_{i,h}(x) + r_{i,h}(x,\mu)$, where $\|r_{i,h}(x,\mu)\| \le c_r\mu^2$, and $\mathbb{E}\big[\|q_{i,h}(x;u,\zeta)\|^2\big] \le Q^2$.

(ii) For each local step (k,t), it holds that $\mathbb{E}\Big[v_{i,k}^t \mid \mathcal{F}_{k,t}\Big] = \mu_k\nabla f_i(\phi_{i,k}^t) + r_{i,k}^t$, where $\|r_{i,k}^t\| \le c_r\mu_k^2$, while $m_i^{-1}\sum_{h=1}^{m_i}\mathbb{E}\Big[\|a_{i,h,k}^t\|^2\Big] \le Q^2$ and $\mathbb{E}\Big[\|v_{i,k}^t\|^2\Big] \le 9Q^2$.

Lemma 1(i) characterizes the conditional mean, smoothing remainder, and second moment of a component query. Lemma 1(ii) shows that the memory correction preserves the conditional mean of fresh single-point estimates with a controlled second moment. Section 3.3 incorporates this direction into the local training ADMM recursion.

Proof of Lemma 1. For (i), apply smoothness at the $\mathcal{F}$-measurable point x:

$$f_{i,h}(x+\mu u) = f_{i,h}(x) + \mu\langle\nabla f_{i,h}(x), u\rangle + R(x,u,\mu), \tag{13}$$

$$|R(x,u,\mu)| \le \frac{L_f\mu^2}{2}\|u\|^2. \tag{14}$$

Assumption 3 gives $\mathbb{E}[u \mid \mathcal{F}] = 0$, $\mathbb{E}[uu^\top \mid \mathcal{F}] = c_u I_d$, and $\mathbb{E}[u\zeta \mid \mathcal{F}] = \mathbb{E}[u\,\mathbb{E}[\zeta \mid \mathcal{F}, u] \mid \mathcal{F}] = 0$. Hence the constant and noise terms vanish, while the linear term yields $\mu\nabla f_{i,h}(x)$. The remainder satisfies

$$r_{i,h}(x,\mu) = \frac{1}{c_u}\mathbb{E}[uR(x,u,\mu) \mid \mathcal{F}],$$

$$\|r_{i,h}(x,\mu)\| \le \frac{L_f\mu^2}{2c_u}\mathbb{E}[\|u\|^3 \mid \mathcal{F}] \le c_r\mu^2.$$

Moreover, (3) and the query-value moment bound imply

$$\mathbb{E}[\|q_{i,h}(x;u,\zeta)\|^2] \le \frac{R_u^2}{c_u^2}\mathbb{E}[|\widetilde{f}_{i,h}(x+\mu u)|^2] \le \frac{R_u^2 M_2}{c_u^2} = Q^2.$$

For (ii), condition on $\mathcal{F}_{k,t}$, which fixes the memory table. For the expectation calculation, couple potential queries $\{q_{i,h,k}^t\}_{h=1}^{m_i}$ independently of the current batch selection; only sampled queries are evaluated. Uniform sampling in (5) gives

$$\begin{aligned}\mathbb{E}[v_{i,k}^t \mid \mathcal{F}_{k,t}] &= \frac{1}{m_i}\sum_{h=1}^{m_i}\mathbb{E}[q_{i,h,k}^t \mid \mathcal{F}_{k,t}] \\ &= \mu_k\nabla f_i(\phi_{i,k}^t) + r_{i,k}^t.\end{aligned} \tag{15}$$

since the sampled-memory mean cancels $\bar{a}_{i,k}^t$. Part (i) bounds each component remainder, and thus $\|r_{i,k}^t\| \le c_r\mu_k^2$.

Each initialized entry has second moment at most Q^2. An entry is refreshed with probability b_i/m_i, independently of its current value. Therefore,

$$\mathbb{E}[\|a_{i,h,k}^{t+1}\|^2] \le \left(1 - \frac{b_i}{m_i}\right)\mathbb{E}[\|a_{i,h,k}^t\|^2] + \frac{b_i}{m_i}Q^2.$$

Induction gives $\mathbb{E}[\|a_{i,h,k}^t\|^2] \le Q^2$ for each h and t. This is an unconditional bound on the stored entries. Uniform sampling, Jensen's inequality, and total expectation then yield

$$\mathbb{E}\left[\left\|\frac{1}{b_i}\sum_{h\in\mathcal{B}_{i,k}^t} q_{i,h,k}^t\right\|^2\right] \le Q^2,$$

$$\mathbb{E}\left[\left\|\frac{1}{b_i}\sum_{h\in\mathcal{B}_{i,k}^t} a_{i,h,k}^t\right\|^2\right] \le Q^2, \qquad \mathbb{E}[\|\bar{a}_{i,k}^t\|^2] \le Q^2.$$

Applying $\|x-y+z\|^2 \le 3(\|x\|^2 + \|y\|^2 + \|z\|^2)$ to (5) gives $\mathbb{E}[\|v_{i,k}^t\|^2] \le 9Q^2$, completing the proof. □

C Message Recursion and Transcript Privacy

Proof of Lemma 2. For an oriented edge $e = (i, j)$, the message rules (9)–(10) give

$$z_{ij,k+1} = \frac{1}{2}\left(z_{ij,k} - z_{ji,k} + 2\rho x_{j,k+1}\right),$$
$$z_{ji,k+1} = \frac{1}{2}\left(z_{ji,k} - z_{ij,k} + 2\rho x_{i,k+1}\right). \tag{16}$$

Their difference gives the ω_k recursion in Lemma 2, and their sum gives

$$z_{ij,k+1} + z_{ji,k+1} = \rho(x_{i,k+1} + x_{j,k+1}). \tag{17}$$

The same sum identity holds at $k = 0$, since $x_{i,0} = x_{j,0} = x_0$ and $z_{ij,0} = z_{ji,0} = \rho x_0$. Consequently,

$$z_{ij,k} = B_{i,e}\omega_{e,k} + \frac{\rho}{2}(x_{i,k} + x_{j,k}). \tag{18}$$

Substitution into (6) yields

$$p_k = \frac{\rho}{2} L_{\mathcal{G}} x_k - B\omega_k. \tag{19}$$

Using $\omega_k = \omega_{k-1} - (\rho/2)B^\top x_k$ in (19) proves the penalty identity in Lemma 2. The definition $\lambda_k = -B\omega_{k-1}$ then gives the equivalent identities in Lemma 2. At initialization, the shifted identity holds due to $\omega_{-1} = \omega_0 = 0$ and $B^\top(\mathbf{1} \otimes x_0) = 0$. Finally, $\lambda_k \in \mathrm{range}(B)$, $\mathbf{1}^\top B = 0$, and $\mathbf{1}^\top L_{\mathcal{G}} = 0$ imply the zero-sum identities. □

Proof of Theorem 1. Fix client i and condition on the transcript history $\mathcal{H}_k$. The vector $c_{i,k} := x_{i,k} - \tau_i\eta_k\mu_k\beta p_{i,k}$ is then fixed. Let $\mathsf{Q}_{\mathcal{D}_i,k}$ be the conditional law of $\bar{s}_{i,k}$. Independence of the release noise gives

$$\mathsf{P}_{\mathcal{D}_i,k} = \int \mathcal{N}(c_{i,k} + s, \sigma^2_{\mathrm{dp},k} I_d)\, \mathsf{Q}_{\mathcal{D}_i,k}(\mathrm{d}s). \tag{20}$$

Both mixing laws under adjacent datasets are supported on $\{s : \|s\| \le R_k^{\mathrm{clip}}\}$, so

$$\|s - s'\| \le 2R_k^{\mathrm{clip}}. \tag{21}$$

For any Rényi order $\gamma > 1$, use the common product measure $\mathsf{Q}_{\mathcal{D}_i,k} \otimes \mathsf{Q}_{\mathcal{D}'_i,k}$ to couple the mixture centers. Data processing and the log-sum inequality give

$$D_\gamma(\mathsf{P}_{\mathcal{D}_i,k} \| \mathsf{P}_{\mathcal{D}'_i,k}) \le \sup_{s,s'} D_\gamma\Big(\mathcal{N}(c_{i,k} + s, \sigma^2_{\mathrm{dp},k} I_d) \Big\| \mathcal{N}(c_{i,k} + s', \sigma^2_{\mathrm{dp},k} I_d)\Big). \tag{22}$$

The equal-covariance Gaussian formula and (21) imply

$$D_\gamma(\mathsf{P}_{\mathcal{D}_i,k} \| \mathsf{P}_{\mathcal{D}'_i,k}) \le \frac{\gamma(2R_k^{\mathrm{clip}})^2}{2\sigma^2_{\mathrm{dp},k}} = \frac{2\gamma(R_k^{\mathrm{clip}})^2}{\sigma^2_{\mathrm{dp},k}}. \tag{23}$$

Thus, client i's conditional release is $2(R_k^{\mathrm{clip}})^2/\sigma^2_{\mathrm{dp},k}$-zCDP.

All outgoing messages are deterministic post-processing of the released state and history. Under client-i adjacency, the other clients' datasets are fixed; their responses use the transcript and fresh independent randomness, adding no separate privacy charge for client i. Adaptive composition yields the cumulative zCDP budget, whose conversion to approximate DP gives the stated privacy guarantee. The adjacency relation replaces the entire local dataset, establishing the stated client-level guarantee. □

D Schur Stability and Disagreement Bounds

Under $0 < a\rho\lambda_N < 8/3$, let $P_\lambda \succ 0$ solve $M_\lambda^\top P_\lambda M_\lambda - P_\lambda = -I_2$. Define $\underline{p} := \min_{2\le\ell\le N} \lambda_{\min}(P_{\lambda_\ell})$ and $\overline{p} := \max_{2\le\ell\le N} \lambda_{\max}(P_{\lambda_\ell})$, and set $\chi := 1/(2\overline{p})$ and $c_w := \overline{p}(2\overline{p} - 1)$. Let $U_\perp$ contain orthonormal eigenvectors for the nonzero Laplacian eigenvalues, and write $\xi_k := \mathrm{col}\big((U_\perp^\top \otimes I_d)x_k,\, (U_\perp^\top \otimes I_d)\lambda_k\big)$. Let S reorder ξ_k into $2d$-dimensional primal–dual pairs indexed by eigenmode. The Lyapunov matrix and energy are $\mathcal{P} := \mathsf{S}^\top \mathrm{diag}_{\ell=2}^N(P_{\lambda_\ell} \otimes I_d)\mathsf{S}$ and

$\mathcal{V}_k := \frac{1}{N}\mathbb{E}[\xi_k^\top \mathcal{P}\xi_k]$. This Lyapunov energy measures how network contraction competes with perturbations from local updates and private releases.

Lemma 3 (Schur stability and input bound)**.** *The matrices M_λ are Schur stable for all nonzero graph modes if and only if $0 < a\rho\lambda_N < 8/3$ holds. Under this condition, the Lyapunov equation has a unique positive-definite solution for each mode, and, for each round k, $\mathcal{V}_{k+1} \le (1-\chi)\mathcal{V}_k + c_w\big[18\eta^2\tau^2Q^2 + 2E_k^{\mathrm{cl}} + \big(1-\frac{1}{N}\big)d\sigma_{\mathrm{dp},k}^2\big]$. Furthermore, $\mathcal{C}_k \le \frac{\mathcal{V}_k}{\underline{p}}$ and $\frac{1}{N}\mathbb{E}[\|p_k\|^2] \le \frac{2(\rho^2\lambda_N^2+1)}{\underline{p}}\mathcal{V}_k$. The common initialization gives $\mathcal{V}_0 = 0$.*

Proof of Lemma 3. Summing (7) and applying (8) gives

$$x_{k+1} = x_k - ap_k + w_k, \tag{24}$$

$$w_k = -\eta\sum_{t=0}^{\tau-1} v_k^t + e_k^{\mathrm{cl}} + \nu_k. \tag{25}$$

The primal–dual identities in Lemma 2 then yield the modal recursion

$$\begin{bmatrix}\widehat{x}_{\lambda,k+1}\\ \widehat{\lambda}_{\lambda,k+1}\end{bmatrix} = M_\lambda\begin{bmatrix}\widehat{x}_{\lambda,k}\\ \widehat{\lambda}_{\lambda,k}\end{bmatrix} + \begin{bmatrix}\widehat{w}_{\lambda,k}\\ 0\end{bmatrix}. \tag{26}$$

The trace and determinant are

$$\mathrm{tr}(M_\lambda) = 2 - a\rho\lambda, \qquad \det(M_\lambda) = 1 - \frac{a\rho\lambda}{2}. \tag{27}$$

The second-order Jury criterion gives

$$\begin{aligned} 1 - \det(M_\lambda) &= \frac{a\rho\lambda}{2},\\ 1 - \mathrm{tr}(M_\lambda) + \det(M_\lambda) &= \frac{a\rho\lambda}{2},\\ 1 + \mathrm{tr}(M_\lambda) + \det(M_\lambda) &= 4 - \frac{3a\rho\lambda}{2}. \end{aligned} \tag{28}$$

These three expressions are positive exactly when $0 < a\rho\lambda < 8/3$. Requiring this for each nonzero Laplacian eigenvalue gives $0 < a\rho\lambda_N < 8/3$.

For a Schur-stable M_λ, the convergent series

$$P_\lambda = \sum_{r=0}^{\infty}(M_\lambda^r)^\top M_\lambda^r \tag{29}$$

is the unique positive-definite solution of the Lyapunov equation. In the ordering of ξ_k, set

$$M := \mathsf{S}^\top \mathrm{diag}_{\ell=2}^{N}(M_{\lambda_\ell}\otimes I_d)\mathsf{S}.$$

Then $M^\top\mathcal{P}M - \mathcal{P} = -I_{2(N-1)d}$. Writing $\|\xi\|_{\mathcal{P}}^2 := \xi^\top\mathcal{P}\xi$, we obtain

$$\|M\xi\|_{\mathcal{P}}^2 = \|\xi\|_{\mathcal{P}}^2 - \|\xi\|^2 \le \left(1 - \frac{1}{\overline{p}}\right)\|\xi\|_{\mathcal{P}}^2. \tag{30}$$

Since $\overline{p} > 1$, Young's inequality with parameter $[2(\overline{p}-1)]^{-1}$ yields

$$\begin{aligned}\|M\xi + \mathrm{col}(\widehat{w},0)\|_{\mathcal{P}}^2 \le\ & \left(1 - \frac{1}{2\overline{p}}\right)\|\xi\|_{\mathcal{P}}^2\\ &+ \overline{p}(2\overline{p}-1)\|\widehat{w}\|^2.\end{aligned} \tag{31}$$

Here $\widehat{w} = (U_\perp^\top\otimes I_d)w$, so $\|\widehat{w}\| = \|\Pi w\|$.

The release noise is independent and zero-mean, eliminating its cross terms with the local update and clipping error. Since Π is nonexpansive, Cauchy–Schwarz and Lemma 1(ii) give

$$\begin{aligned}\frac{1}{N}\mathbb{E}[\|\Pi w_k\|^2] &\le \frac{2\eta^2\tau}{N}\sum_{t=0}^{\tau-1}\mathbb{E}[\|v_k^t\|^2] + 2E_k^{\mathrm{cl}}\\ &\quad + \left(1 - \frac{1}{N}\right)d\sigma_{\mathrm{dp},k}^2\\ &\le 18\eta^2\tau^2Q^2 + 2E_k^{\mathrm{cl}} + \left(1 - \frac{1}{N}\right)d\sigma_{\mathrm{dp},k}^2.\end{aligned} \tag{32}$$

Taking expectations in (31) and dividing by N proves the Lyapunov recursion in Lemma 3.

The eigenvalue bounds on $\mathcal{P}$ imply the consensus bound in Lemma 3. Using $p_k = \rho L_{\mathcal{G}} x_k + \lambda_k$ further gives

$$\frac{1}{N}\mathbb{E}[\|p_k\|^2] \le 2\rho^2\lambda_N^2\mathcal{C}_k + \frac{2}{N}\mathbb{E}[\|\lambda_k\|^2] \le \frac{2(\rho^2\lambda_N^2+1)}{\underline{p}}\mathcal{V}_k,$$

which proves the penalty bound in Lemma 3. The common initialization has zero primal disagreement and $\lambda_0 = 0$, hence $\mathcal{V}_0 = 0$. □

Remark 1. The modal analysis describes coordination among $N \ge 2$ clients. For $N = 1$, the graph correction and dual state vanish, so $p_k = \lambda_k = 0$. Both disagreement measures are identically zero because the network average equals the sole client's iterate. The algorithm then performs local derivative-free updates with clipping and Gaussian noise applied at each release. Its descent estimate follows directly from smoothness and the oracle bounds in Lemma 1, without introducing graph eigenvalues or modal Lyapunov constants.

E Descent and Convergence Bounds

We use the synchronized setting of Section 4.2. Define the within-round disagreement as $\mathcal{C}_{k,t}^{\phi} := \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[\|\phi_{i,k}^t - \bar{\phi}_k^t\|^2]$.

To bound within-round drift, define $\Gamma := \frac{3+6(\tau-1)^2\alpha^2\beta^2(\rho^2\lambda_N^2+1)}{\underline{p}}$. The next estimate transfers control of the released-state disagreement to the local iterates between communication rounds.

Lemma 4 (Within-round disagreement)**.** *Under $0 < a\rho\lambda_N < 8/3$, for each round k and $t = 0, \dots, \tau-1$, $\mathcal{C}_{k,t}^{\phi} \le \Gamma\mathcal{V}_k + 27\eta^2t^2Q^2$.*

Proof of Lemma 4. Iterating (7) gives

$$\phi_k^t = x_k - \eta\sum_{s=0}^{t-1} v_k^s - t\alpha\beta p_k. \tag{33}$$

Since $\Pi p_k = p_k$, applying Π and $\|a+b+c\|^2 \le 3(\|a\|^2+\|b\|^2+\|c\|^2)$ yields, by Lemma 1(ii),

$$\mathcal{C}_{k,t}^{\phi} \le 3\mathcal{C}_k + 27\eta^2t^2Q^2 + 3t^2\alpha^2\beta^2\frac{1}{N}\mathbb{E}\big[\|p_k\|^2\big]. \tag{34}$$

The consensus and penalty bounds in Lemma 3, together with $t \le \tau - 1$, yield the estimate in Lemma 4. □

For the descent bound, collect the local-query and smoothing terms in $A_{\mathrm{loc}} := \frac{9L_f}{2}\tau\eta^2Q^2 + \tau\alpha c_r^2\mu^2 + \frac{9}{8}\alpha L_f^2\eta^2Q^2\big[2\tau(\tau-1)(2\tau-1)+1\big]$. Applying smoothness with the oracle and disagreement bounds gives the objective decrease over one communication round.

Lemma 5 (One-round descent)**.** *Under the standing assumptions and $0 < a\rho\lambda_N < 8/3$, for each round k, $\mathbb{E}[F(\bar{x}_{k+1})] \le \mathbb{E}[F(\bar{x}_k)] - \frac{\alpha}{8}\sum_{t=0}^{\tau-1}\mathbb{E}[\|\nabla F(\bar{\phi}_k^t)\|^2] + \frac{\alpha L_f^2\tau\Gamma}{2}\mathcal{V}_k + A_{\mathrm{loc}} + \big(\frac{4}{\alpha}+\frac{L_f}{2}\big)E_k^{\mathrm{cl}} + \frac{L_f}{2N}d\sigma_{\mathrm{dp},k}^2$.*

Proof of Lemma 5. The zero-sum identity $\mathbf{1}^\top p_k = 0$ gives

$$\bar{\phi}_k^{t+1} = \bar{\phi}_k^t - \eta\bar{v}_k^t, \qquad \bar{v}_k^t = \frac{1}{N}\sum_{i=1}^{N} v_{i,k}^t. \tag{35}$$

Set $G_k^t := \nabla F(\bar{\phi}_k^t)$ and $g_k^t := N^{-1}\sum_i \nabla f_i(\phi_{i,k}^t)$. Smoothness and Jensen's inequality imply

$$\mathbb{E}\big[\|g_k^t - G_k^t\|^2\big] \le L_f^2\mathcal{C}_{k,t}^{\phi}. \tag{36}$$

Lemma 1(ii) also gives

$$\mathbb{E}\big[\bar{v}_k^t \mid \mathcal{F}_{k,t}\big] = \mu g_k^t + \bar{r}_k^t, \qquad \|\bar{r}_k^t\| \le c_r\mu^2. \tag{37}$$

where $\bar{r}_k^t := N^{-1}\sum_i r_{i,k}^t$. By smoothness, conditional expectation, and $\mathbb{E}[\|\bar{v}_k^t\|^2] \le 9Q^2$,

$$\mathbb{E}\big[F(\bar{\phi}_k^{t+1})\big] \le \mathbb{E}\big[F(\bar{\phi}_k^t)\big] - \alpha\mathbb{E}\big[\langle G_k^t, g_k^t\rangle\big] - \eta\mathbb{E}\big[\langle G_k^t, \bar{r}_k^t\rangle\big] + \frac{9L_f}{2}\eta^2Q^2. \tag{38}$$

The inequalities

$$\langle G_k^t, g_k^t\rangle \ge \frac{1}{2}\|G_k^t\|^2 - \frac{1}{2}\|g_k^t - G_k^t\|^2,$$
$$\eta|\langle G_k^t, \bar{r}_k^t\rangle| \le \frac{\alpha}{4}\|G_k^t\|^2 + \alpha c_r^2\mu^2 \tag{39}$$

therefore yield

$$\mathbb{E}\big[F(\bar{\phi}_k^{t+1})\big] \le \mathbb{E}\big[F(\bar{\phi}_k^t)\big] - \frac{\alpha}{4}\mathbb{E}\big[\|G_k^t\|^2\big] + \frac{\alpha L_f^2}{2}\mathcal{C}_{k,t}^{\phi} + \alpha c_r^2\mu^2 + \frac{9L_f}{2}\eta^2Q^2. \tag{40}$$

Sum over t and use Lemma 4. Since $\sum_{t=0}^{\tau-1} t^2 = \tau(\tau-1)(2\tau-1)/6$, the definition of A_{loc} gives

$$\mathbb{E}[F(\bar{\phi}_k^{\tau})] \le \mathbb{E}[F(\bar{x}_k)] - \frac{\alpha}{4}\sum_{t=0}^{\tau-1}\mathbb{E}[\|G_k^t\|^2] + \frac{\alpha L_f^2\tau\Gamma}{2}\mathcal{V}_k + A_{\mathrm{loc}} - \frac{9}{8}\alpha L_f^2\eta^2Q^2. \tag{41}$$

The average release satisfies

$$\bar{x}_{k+1} = \bar{\phi}_k^{\tau} + \bar{e}_k^{\mathrm{cl}} + \bar{\nu}_k. \tag{42}$$

Conditionally on the pre-noise variables, $\bar{\nu}_k$ is zero-mean and

$$\mathbb{E}\big[\|\bar{\nu}_k\|^2\big] = \frac{d\sigma_{\mathrm{dp},k}^2}{N}. \tag{43}$$

A second application of smoothness yields

$$\mathbb{E}[F(\bar{x}_{k+1})] \le \mathbb{E}\big[F(\bar{\phi}_k^{\tau})\big] + \mathbb{E}\big[\langle\nabla F(\bar{\phi}_k^{\tau}), \bar{e}_k^{\mathrm{cl}}\rangle\big] + \frac{L_f}{2}\mathbb{E}\big[\|\bar{e}_k^{\mathrm{cl}}\|^2\big] + \frac{L_f}{2N}d\sigma_{\mathrm{dp},k}^2. \tag{44}$$

Young's inequality gives

$$\langle\nabla F(\bar{\phi}_k^{\tau}), \bar{e}_k^{\mathrm{cl}}\rangle \le \frac{\alpha}{16}\|\nabla F(\bar{\phi}_k^{\tau})\|^2 + \frac{4}{\alpha}\|\bar{e}_k^{\mathrm{cl}}\|^2. \tag{45}$$

Since $\bar{\phi}_k^{\tau} = \bar{\phi}_k^{\tau-1} - \eta\bar{v}_k^{\tau-1}$, smoothness also gives

$$\mathbb{E}\big[\|\nabla F(\bar{\phi}_k^{\tau})\|^2\big] \le 2\mathbb{E}\big[\|G_k^{\tau-1}\|^2\big] + 18L_f^2\eta^2Q^2. \tag{46}$$

Finally, Jensen's inequality yields $\mathbb{E}[\|\bar{e}_k^{\mathrm{cl}}\|^2] \le E_k^{\mathrm{cl}}$. Substituting (45)–(46) into (44) adds at most $(\alpha/8)\mathbb{E}[\|G_k^{\tau-1}\|^2]$ to the gradient terms. The accompanying $(9/8)\alpha L_f^2\eta^2Q^2$ cancels the last term in (41). Bounding each remaining gradient coefficient by $-\alpha/8$ proves the one-round descent bound in Lemma 5. □

Proof of Theorem 2. Define

$$\mathcal{R}_K := 18K\eta^2\tau^2Q^2 + 2\mathcal{E}_{\mathrm{cl},K} + \left(1 - \frac{1}{N}\right)\mathcal{E}_{\mathrm{dp},K}.$$

Summing the Lyapunov recursion in Lemma 3 and using $\mathcal{V}_K \ge 0$ gives

$$\chi\sum_{k=0}^{K-1}\mathcal{V}_k \le \mathcal{V}_0 + c_w\mathcal{R}_K. \tag{47}$$

The consensus bound in Lemma 3 then implies

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathcal{C}_k \le \frac{\mathcal{V}_0 + c_w\mathcal{R}_K}{K\chi\underline{p}}. \tag{48}$$

Summing the descent bound in Lemma 5 and using $F(\bar{x}_K) \geq F_\star$ yields

$$\frac{\alpha}{8}\sum_{k=0}^{K-1}\sum_{t=0}^{\tau-1}\mathbb{E}[\|\nabla F(\bar{\phi}_k^t)\|^2] \leq F(\bar{x}_0) - F_\star + KA_{\mathrm{loc}} + \frac{\alpha L_f^2 \tau\Gamma}{2}\sum_{k=0}^{K-1}\mathcal{V}_k + \left(\frac{4}{\alpha} + \frac{L_f}{2}\right)\mathcal{E}_{\mathrm{cl},K} + \frac{L_f}{2N}\mathcal{E}_{\mathrm{dp},K}. \tag{49}$$

Substitute (47), divide by $K\tau\alpha/8$, and add (48). This gives

$$\mathcal{S}_K \leq \frac{8[F(\bar{x}_0) - F_\star]}{K\tau\alpha} + \frac{8A_{\mathrm{loc}}}{\tau\alpha} + \left(4L_f^2\Gamma + \frac{1}{\underline{p}}\right)\frac{\mathcal{V}_0 + c_w\mathcal{R}_K}{K\chi} + \left(\frac{32}{\tau\alpha^2} + \frac{4L_f}{\tau\alpha}\right)\overline{\mathcal{E}}_{\mathrm{cl},K} + \frac{4L_f}{N\tau\alpha}\overline{\mathcal{E}}_{\mathrm{dp},K}. \tag{50}$$

The common initialization gives $\mathcal{V}_0 = 0$. For fixed $a = \tau\beta\alpha$, ρ, and graph,

$$(\tau-1)^2\alpha^2\beta^2 = \left(\frac{\tau-1}{\tau}\right)^2 a^2 \leq a^2.$$

Thus, Γ is bounded uniformly over τ, and the Lyapunov constants are fixed. Directly from their definitions,

$$\frac{A_{\mathrm{loc}}}{\tau\alpha} = \mathcal{O}\left(\frac{\eta}{\mu} + \mu^2 + \eta^2\tau^2\right),$$
$$\frac{\mathcal{R}_K}{K} = \mathcal{O}\left(\eta^2\tau^2 + \overline{\mathcal{E}}_{\mathrm{cl},K} + \overline{\mathcal{E}}_{\mathrm{dp},K}\right).$$

For $\alpha > 0$ and $\tau \geq 1$, $1/(\tau\alpha) \leq 1 + 1/(\tau\alpha^2)$. Substitution into (50) establishes (11). □

Proof of Corollary 1. The nonprivate parameter schedule fixes $a = \tau\beta_{T,\tau}\eta_T\mu_T = a_0$. Since $T = K\tau$, each of $1/(T\eta_T\mu_T)$, η_T/μ_T, and μ_T^2 is $\mathcal{O}(T^{-1/3})$, while $\eta_T^2\tau^2 = \mathcal{O}(\tau^2/T)$. The release terms vanish, so (11) gives (12). The stated local-update range and communication count follow from $\tau^2/T = \mathcal{O}(T^{-1/3})$ and $K = T/\tau$. □

Uniform constants and transcript accounting. The constants in the comparison in Section 4.4 remain controlled as the local-stage length grows. For fixed ρ and graph, the nonprivate parameter schedule maintains $a = a_0$, so the modal matrices M_λ and their Lyapunov solutions P_λ are unchanged. Since $(\tau-1)^2\alpha^2\beta^2 \leq a_0^2$, the coefficient Γ is uniformly bounded in T and τ. The uniform oracle-moment condition in Corollary 1 also makes the bound $9Q^2$ in Lemma 1(ii) independent of T and τ. Applying the nonprivate parameter schedule to Theorem 2 gives (12) with uniform constants and the explicit local-drift term τ^2/T. The privacy calculation uses the same round-level description. Each client forms its outgoing messages by deterministic post-processing of one protected state and the preceding transcript, as established in Theorem 1. Under replacement of one client's dataset, privacy composes over that client's K releases, whereas the communication count includes all $2|\mathcal{E}|K$ directed messages. The other clients' conditional response laws are unchanged under this replacement, so adaptive composition covers the full interactive transcript.

The following remark addresses convex objectives.

Remark 2. For convex objectives, Theorem 2 still bounds stationarity and disagreement under the stated assumptions. If a minimizer $x^\star$ exists, convexity gives $F(x) - F(x^\star) \leq \langle\nabla F(x), x - x^\star\rangle$, so exact stationarity implies global optimality. Turning this relation into an objective-gap rate requires additional control of the distance to the solution set. In particular, Corollary 1 preserves the stationarity and consensus rate with the same reduction in communication rounds for convex instances.

Communication and privacy trade-offs. SPADE-DFL characterizes how local computation can replace neighbor communication when learning relies on single-point function evaluations. For smooth nonconvex objectives under uniform query-moment bounds, the prescribed nonprivate schedule yields a time-averaged stationarity and consensus bound of $\mathcal{O}(T^{-1/3} + \tau^2/T)$, where

T is the number of local updates per client and τ is the number of updates between exchanges. This dependence permits $\tau = \Theta(T^{1/3})$, reducing communication to $\Theta(T^{2/3})$ rounds while preserving the $\mathcal{O}(T^{-1/3})$ convergence order. The communication interval also determines how often local information is privately released. Protecting the accumulated data-dependent increment once per round establishes client-level differential privacy for the full interactive transcript. With a fixed clipping radius and total zCDP budget, fewer exchanges reduce the required uniform Gaussian noise variance per release. The finite-horizon bound quantifies the accompanying local drift, clipping, and release errors, making explicit how the choice of communication interval affects the optimization cost of private training.

F Datasets, Models, and Data Sources

We use four datasets, i.e., MNIST, Fashion-MNIST, Alzheimer's disease, and Brain Tumor MRI. The MNIST task distinguishes digits 6 and 7, and the Fashion-MNIST task distinguishes T-shirt/top from Trouser. The Alzheimer's disease task uses structured clinical features for binary classification. Brain Tumor MRI is used for four-class image classification.

For MNIST and Fashion-MNIST, the original 784-dimensional image vectors are reduced to 10 principal components before training. The Alzheimer's disease task uses 39-dimensional structured clinical features. For Brain Tumor MRI, a frozen ResNet-18 V1 network extracts 512-dimensional representations. Principal component analysis fitted on the training split reduces these representations to 10 dimensions, followed by a four-class linear softmax classifier with 40 trainable parameters.

Four client-data distributions are considered: independent and identically distributed (IID), Dirichlet partitions with concentration parameters $\alpha = 0.6$ and $\alpha = 0.3$, and a pathological partition. A smaller Dirichlet concentration produces stronger variation in class proportions across clients. The same data partition and random seed are used across methods within each comparison.

Dataset sources.

- **MNIST.** https://www.kaggle.com/datasets/hojjatk/mnist-dataset
- **Fashion-MNIST.** https://www.kaggle.com/datasets/zalando-research/fashionmnist
- **Alzheimer's disease.** https://www.kaggle.com/datasets/rabieelkharoua/alzheimers-disease-dataset
- **Brain Tumor MRI.** https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

G Additional Method and Experimental Details

G.1 Extended Method Overview and Contributions

The construction separates the accumulated local learning increment from the correction determined by preceding exchanges. Decentralized formulations of the alternating direction method of multipliers (ADMM) express agreement through constraints on neighboring models (Boyd et al., 2011; Shi et al., 2014). Linearization makes the local primal update explicit (Ling et al., 2015), while curvature aided formulations permit local primal and dual steps without inner communication loops (Li et al., 2023). Local training further extends the computation performed between exchanges (Ren et al., 2026a). Reference estimates also provide a way to reuse information in zeroth-order optimization (Gautam et al., 2024). Building on these ideas, we construct local directions from a component memory that preserves the conditional mean of fresh single-point estimates. The ADMM correction is held fixed throughout each local stage. An exact message recursion relates accumulated local updates to subsequent network disagreement and supports the joint selection of the local training length and round gain. The placement of privacy protection also matters since clipping can bias stochastic updates (Koloskova et al., 2023). We therefore isolate the accumulated data dependent increment from the correction fixed by the preceding transcript. Clipping this increment before adding calibrated Gaussian noise produces one protected state per client and round from which all

outgoing messages are constructed. Retaining the explicit graph correction makes the effect of each release visible in subsequent exchanges. The message recursion determines how the local stage can grow while preserving the convergence order under the nonprivate parameter schedule. For private training, a finite horizon bound accounts for clipping and Gaussian releases in the stationarity and consensus criterion. The main contributions are as follows.

1. **Communication savings from local loss evaluations.** We establish communication savings from local loss evaluations for smooth nonconvex objectives under uniform query moment control. The prescribed nonprivate schedule yields a time averaged stationarity and consensus bound of $\mathcal{O}(T^{-1/3} + \tau^2/T)$ after $T = K\tau$ local updates per client. The drift term permits $\tau = \Theta(T^{1/3})$ local updates per exchange while retaining the $\mathcal{O}(T^{-1/3})$ order. For a joint tolerance $\varepsilon_{\mathrm{stat}}$, this reduces the sufficient communication bound from $\mathcal{O}(\varepsilon_{\mathrm{stat}}^{-3})$ for the same method with $\tau = 1$ to $\mathcal{O}(\varepsilon_{\mathrm{stat}}^{-2})$ rounds.
2. **Single-point local training and network dynamics.** SPADE-DFL incorporates a component-memory single-point estimator into local training ADMM. Under uniform query-moment bounds, the memory correction preserves the conditional mean of fresh single-point estimates and admits a controlled second moment. An exact message recursion yields a modal representation of network disagreement. In the synchronized setting, the homogeneous disagreement modes are Schur stable exactly when $0 < \tau\beta\eta\mu\rho\lambda_N < 8/3$, where λ_N is the largest graph Laplacian eigenvalue. This characterization determines the admissible round gain used in the local training schedule.
3. **Protecting local computation through the state used for coordination.** We isolate the accumulated private update from the graph correction fixed by the observed history. One Gaussian release per client and round protects the clipped update, while all outgoing messages follow by deterministic processing. We prove differential privacy for the full interactive transcript under replacement of an entire client dataset, accounting for effects propagated through subsequent neighbor responses. A finite horizon analysis traces release perturbations through the feedback governing objective descent. The resulting bound quantifies the optimization cost of privacy under local drift, linking information disclosed during communication to collective learning accuracy.

G.2 Equivalent Updates and Release Operations

Repeated neighbor synchronization can dominate the cost of derivative-free decentralized learning. To reduce the communication frequency, SPADE-DFL freezes the ADMM correction during a local stage, performs multiple single-query updates, and communicates only after the stage is completed. Consequently, client i executes τ_i local updates between two consecutive neighbor exchanges, which is illustrated in Algorithm 1. Round k uses step size $\eta_k > 0$ and smoothing radius $\mu_k > 0$, while $\rho, \beta > 0$ determine the ADMM correction. All clients use $x_{i,0} = x_0$ and $z_{ij,0} = \rho x_0$ for every $j \in \mathcal{N}_i$, hence $z_{ij,0} + z_{ji,0} = \rho(x_{i,0} + x_{j,0})$ on each edge. The initialization, graph, and parameter schedules are public and data-independent. Fresh local randomization is independent across clients, and release noise is independent of all pre-release variables.

Equivalent local updates. The local step in (7) admits the following rescaling and accumulated form. Let $\alpha_k = \eta_k\mu_k$ and $\widehat{v}_{i,k}^t = v_{i,k}^t/\mu_k$. The local update equivalently reads $\phi_{i,k}^{t+1} = \phi_{i,k}^t - \alpha_k\left(\widehat{v}_{i,k}^t + \beta p_{i,k}\right)$. Thus, α_k is the effective descent stepsize and $\alpha_k\beta$ is the coefficient of the frozen ADMM correction per local step. Under the synchronized parameters settings, $a = \tau\alpha\beta$ is the aggregate network gain over one communication round. After τ_i updates, define $y_{i,k+1} := \phi_{i,k}^{\tau_i}$ and $s_{i,k} := -\eta_k \sum_{t=0}^{\tau_i-1} v_{i,k}^t$. Since $p_{i,k}$ is fixed within the round, it follows that $y_{i,k+1} = x_{i,k} - \tau_i\eta_k\mu_k\beta p_{i,k} + s_{i,k}$. Therefore, the state $y_{i,k+1}$ is intermediate and unreleased.

Protected release and message formation. Before communication, SPADE-DFL sanitizes the accumulated data-dependent update $s_{i,k}$ once, while leaving the explicit ADMM correction unchanged. The release noise is independent of all variables generated before the corresponding release and is conditionally independent across clients and rounds. Specifically, client i projects the accumulated data-dependent update $s_{i,k}$ onto the closed Euclidean ball centered at the origin

with radius R_k^{clip}, and denotes the resulting clipped vector by $\bar{s}_{i,k}$. It then samples a zero-mean Gaussian vector $\nu_{i,k}$ with covariance $\sigma_{\mathrm{dp},k}^2 I_d$. The private released state is given by (8). Define $e_{i,k}^{\mathrm{cl}} := \bar{s}_{i,k} - s_{i,k}$, which gives the equivalent decomposition $x_{i,k+1} = y_{i,k+1} + e_{i,k}^{\mathrm{cl}} + \nu_{i,k}$. The released state is then used to form each outgoing message by (9). After receiving the corresponding message from neighbor j, client i updates the directed auxiliary state by (10). Client i completes τ_i local updates before each neighbor exchange, with all outgoing messages constructed from the released state $x_{i,k+1}$. Communication is thus amortized over multiple local updates and each directed edge carries one d-dimensional ADMM message per round, the complete workflow of which is summarized in Fig. 1.

G.3 Sensitivity to Client Count and Privacy Budget

The following detailed results supplement Section 5.3 under the partition and privacy conditions stated there. Client counts range from 30 to 100 at $\varepsilon = 32$, as listed in the left block of Table 2. MNIST accuracy remained between 99.03% and 99.21%. The corresponding ranges were 92.23–94.22% for Fashion-MNIST, 72.16–73.00% for Alzheimer's disease, and 65.00–70.96% for Brain Tumor MRI. Brain Tumor MRI showed the largest variation and reached its highest mean accuracy at $N = 100$. Privacy budgets ε range from 4 to 32 at $N = 100$, as detailed in the right block of Table 2. Mean validation accuracy increased from 93.32% to 99.15% on MNIST, from 92.50% to 93.55% on Fashion-MNIST, from 61.05% to 73.00% on Alzheimer's disease, and from 43.29% to 70.96% on Brain Tumor MRI. The largest gain occurred on Brain Tumor MRI. MNIST accuracy changed little beyond $\varepsilon = 16$, while Fashion-MNIST varied by approximately one percentage point over the tested range.