FMUON: COORDINATED MOMENTUM ORTHOGONALIZATION FOR FEDERATED TRAINING OF LARGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

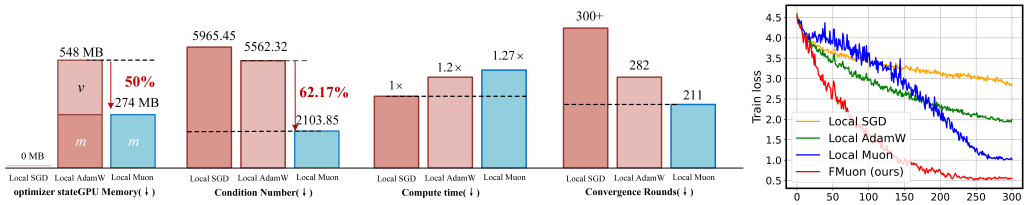
Existing Federated Learning (FL) methods still primarily use element-wise local optimizers (e.g. AdamW, SGD) in each client, neglecting the geometric structure of the weight matrices. This often leads to the amplification of pathological directions in the weights during local updates, leading to deterioration in the condition number and slow convergence. Therefore, we introduce the Muon optimizer in each client (named `Local Muon`), which has matrix orthogonalization to optimize matrix-structured parameters. Experimental results show that, in the IID setting, `Local Muon` significantly accelerates the convergence of FL and reduces communication rounds compared to Local SGD and Local AdamW. However, in the non-IID setting, independent matrix orthogonalization based on the local distributions of each client induces strong client drift. Applying Muon in non-IID FL poses significant challenges: (1) Local–global lmo mismatch leading to client drift; (2) Momentum reinitialization. To address these challenges, we propose a novel Federated Muon optimizer (`FMuon`), which incorporates two key techniques: (1) momentum aggregation, where clients use the aggregated momentum for local initialization; (2) local-global alignment, where the local gradients are aligned with the global update direction to significantly reduce client drift. Theoretically, we establish a convergence guarantee for `FMuon`. Empirically, we validate the effectiveness of `FMuon` on language and vision models. Compared to several baselines including `Local Muon`, `FMuon` significantly reduces communication rounds and improves test accuracy. The code is available in https://anonymous.4open.science/r/FedMuon-935D.

1 INTRODUCTION

With the rapid growth of data and rising concerns over user privacy, traditional centralized training paradigms have become inadequate. Federated Learning (FL) McMahan et al. (2017) offers a scalable and privacy-preserving framework that enables collaborative model training across decentralized clients without sharing raw data (Liu et al., 2024). As data becomes increasingly siloed, FL is a practical solution for large-scale distributed deep learning. However, data heterogeneity and limited communication rounds create significant bottlenecks in FL. Recent studies reveal that the Hessian matrix in neural networks exhibits an approximate block-diagonal structure with several dense sub-blocks (Collobert, 2004; Zhang et al., 2025). Understanding parameter matrix structures is crucial for effective federated aggregation, yet this perspective has been largely overlooked in the federated learning literature. Currently, when clients use element-wise optimizers (such as AdamW/SGD) for multi-step updates on their local data, the weight matrices may gradually become ill-conditioned as shown in Figure 1, causing the update directions to either cancel out or amplify after aggregation. As a result, in each communication round clients struggle to obtain effective updates, and the global model converges slowly.

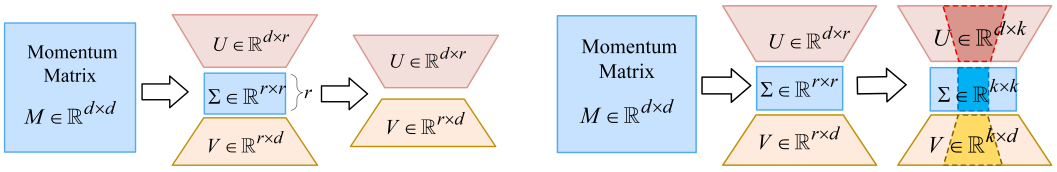
Recent advancements in the Muon optimizer offer a novel solution to this challenge. The Muon optimizer (Jordan et al., 2024) has recently demonstrated that orthogonal normalization of weight update matrices can significantly accelerate neural network training, as shown in Figure 2. By conditioning the weight updates to produce consistent changes in the hidden states, orthogonal normalization updates lead to faster convergence, improved training stability, and better hyperparameter transferability

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.



(a) Comparison of local optimizers in FL. (b) ViT-Tiny, non-IID

Figure 1: (a) Analysis of ViT-Tiny on CIFAR-100, showing optimizer state memory, condition number, computation time, and convergence rounds. Local Muon achieves lower memory cost, lower the condition number, and faster convergence. (b) Training loss curves of ViT-Tiny on non-IID data.



(a) Matrix orthogonalization with SVD (b) Matrix compression using SVD

Figure 2: (a) shows SVD-based matrix orthogonalization; (b) applies SVD to the momentum matrix $M \in \mathbb{R}^{d\times d}$, i.e., $M \approx U_k \Sigma_k V_k^\top$, and keeps the top-$k$ singular vectors to obtain $U \in \mathbb{R}^{d\times k}$ and $V \in \mathbb{R}^{k\times d}$.

across different model scales (Bernstein & Newhouse, 2024; Pethick et al., 2025). Moonshot AI (Liu et al., 2025) found that, when training a 16B model, Muon achieved nearly twice the computational efficiency compared to AdamW (Loshchilov et al., 2017). Similarly, Essential AI (Shah et al., 2025) observed significant improvements with Muon in large-batch training. Both GLM 4.5 and K2 are trained with the Muon optimizer (Liu et al., 2025). These features suggest that using Muon for local training in FL (`Local Muon`) could accelerate local training and reduce communication rounds.

We have also validated the effectiveness of `Local Muon` in FL in the Independent and Identically Distributed (IID) setting. `Local Muon` significantly outperforms Local SGD and Local AdamW as shown in Figure 4. `Local Muon` accelerates local convergence and reduces the number of communication rounds required to reach the same level of precision, with faster local loss decrease, smoother training curves, and faster global model convergence, as shown in Figure 4. *However, in the non-IID setting, although the local losses of each client still decrease rapidly, the global model after aggregation becomes significantly unstable or even fails to converge (see Figure 4).* We identify the reasons why the Muon optimizer fails in non-IID FL from two complementary perspectives.

(Challenge 1) ***Local–global lmo mismatch****: Muon constructs LMO directions from local momentum, whose aggregation need not yield a descent direction for the global objective.*

(Challenge 2) ***Momentum reinitialization:*** *reinitializing the momentum of Muon from scratch in every round hinders the convergence.*

These challenges motivate us to develop a novel **Federated Muon** optimizer, `FMuon`, the FL optimizer that explicitly accounts for the structure of update matrices. `FMuon` addresses the impact of non-IID data through two key mechanisms: (1) local-global alignment, where the current local gradients are aligned with the global update to significantly reduce cross-client inconsistency; (2) momentum aggregation, where clients initialize using the aggregated momentum.

Our contributions are summarized as follows:

- We identify two obstacles to extending `Muon` to heterogeneous federated learning: local–global LMO mismatch and momentum reinitialization across rounds. These expose why effective local matrix optimization need not translate into effective global optimization.
- We propose `FMuon`, which combines local–global alignment with momentum aggregation to address these obstacles. We further develop an SVD-based momentum compression strategy that reduces communication overhead while retaining comparable empirical performance. Extensive vision and language experiments demonstrate the effectiveness of these mechanisms.
- **Establishing vanishing stationarity guarantees.** Under full participation with exact LMOs, we establish non-convex convergence bounds for `Local Muon` and `FMuon`. Local Muon bound retains

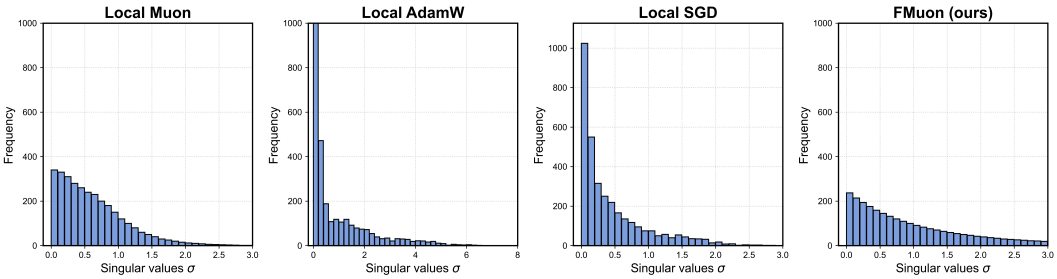
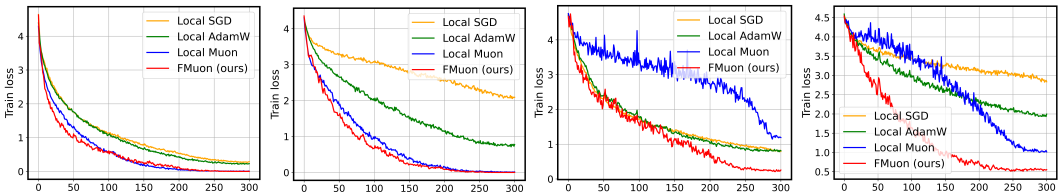


Figure 3: Singular value distributions of `Local Muon`, Local AdamW, Local SGD, and `FMuon`. Local SGD/AdamW are more ill-conditioned with heavier tails and larger singular values, while `FMuon` has a more balanced spectrum and a smaller condition number (where the condition number is defined as the ratio between the largest and smallest singular values).



(a) ResNet-18, IID **(b)** ViT-Tiny, IID **(c)** ResNet-18, non-IID **(d)** ViT-Tiny, non-IID

Figure 4: Performance comparison of Local SGD, Local AdamW, `Local Muon`, and our `FMuon` on CIFAR-100 under IID and non-IID settings.

heterogeneity- and noise-dependent residuals, whereas `FMuon` achieves an $O(R^{-1/4})$ stationarity guarantee with a decaying heterogeneity contribution. This comparison establishes a stronger asymptotic guarantee without interpreting a non-vanishing upper-bound term.

2 Related Work

• **Optimizers in non-IID Federated Learning.** Data heterogeneity across clients is a fundamental challenge in FL. A range of algorithms have been proposed to mitigate the adverse effects of non-IID data distributions. For example, FedProx (Li et al., 2020a) introduces a proximal term to restrict local updates; SCAFFOLD (Karimireddy et al., 2020b) applies control variates to correct client drift; and FedCM (Xu et al., 2021) leverages client momentum to stabilize updates. More recently, Sun et al. (2023) proposed FedLADA to only aggregate the second-momentum estimate of Adam to overcome client drift. **Novelty.** Prior correction methods (e.g., SCAFFOLD, FedCM) assume local SGD and overlook other optimizers. Thérien et al. (2025) proposed MuLoCo, which extends Muon to Local SGD, their method is limited to homogeneous settings and lacks theoretical guarantees. FedMuon(Takezawa et al., 2026) also study Muon in heterogeneous federated learning. Their FedMuon corrects the input to the LMO using SCAFFOLD-style local and global control variates. Our method uses a different update: it applies the LMO to local momentum, combines the resulting direction with the previous global update, and initializes each round with aggregated momentum. We additionally study low-rank compression of the shared momentum and evaluate the method across vision and language models.

• **Optimizers in Centralized Settings.** Although widely used optimizers such as SGD, Adam (Kingma & Ba, 2015), and AdamW (Loshchilov et al., 2017) are effective in many deep learning settings, they generally treat inherently structured parameters (e.g., matrices) as flattened vectors during optimization. In contrast, recent work has increasingly focused on structure-aware optimizers that make explicit use of the underlying parameter geometry. Examples include AdaGrad (Duchi et al., 2011), LAMB (You et al., 2020), and Adam-mini (Zhang et al., 2025), which exploit matrix- or layer-level structure to reduce memory footprint. Shampoo (Gupta et al., 2018) further targets matrix and tensor parameters and can be interpreted as an efficient approximation to AdaGrad's full-matrix preconditioner (Duchi et al., 2011). More recently, SOAP (Vyas et al., 2025) integrates the ideas of Adam with Shampoo's matrix-aware design. The Muon optimizer (Jordan et al.,

2024) extends this line of work by orthogonalizing weight-update matrices, yielding substantially faster and more stable neural network training.

3 CHALLENGES OF MUON IN FL

FL aims to optimize model parameters with local clients, i.e., minimizing the following problem:

$$f_i(\boldsymbol{x}) := \mathbb{E}_{\xi_i \sim \mathcal{D}_i}\big[F_i(\boldsymbol{x}; \xi_i)\big], \quad f(\boldsymbol{x}) := \frac{1}{N}\sum_{i=1}^{N} f_i(\boldsymbol{x}). \tag{1}$$

The function F_i is the loss function on client i. $\mathbb{E}_{\xi_i \sim \mathcal{D}_i}[\cdot]$ denotes conditional expectation with respect to the sample ξ_i. N is the number of clients, and $\boldsymbol{x}$ is global model.

3.1 THE MUON OPTIMIZER

The Muon Optimizer. Muon exploits the matrix structure of model parameters by orthogonalizing momentum before applying an update (Jordan et al., 2024). We describe its basic momentum formulation, omitting weight decay and implementation-specific scaling. For a weight matrix W_{t-1} and gradient estimate G_t, the momentum is $M_t = \beta M_{t-1} + (1-\beta)G_t, \qquad M_0 = 0$. The idealized update admits a linear minimization oracle (LMO) interpretation over the spectral-norm unit ball:

$$Q_t \in \arg\min_{\|Q\|_{\mathrm{sp}} \le 1} \langle M_t, Q\rangle, W_t = W_{t-1} + \eta_t Q_t, \tag{2}$$

where $\langle A, B\rangle = \mathrm{tr}(A^\top B)$. For the compact singular value decomposition $M_t = U_t \Sigma_t V_t^\top$, a canonical minimizer is $Q_t = -U_t V_t^\top$, satisfying $\langle M_t, Q_t\rangle = -\|M_t\|_*$, where $\|\cdot\|_*$ denotes the nuclear norm. We choose $Q_t = 0$ when $M_t = 0$. This transformation preserves the nonzero singular-vector pairs and sets their singular values to one, balancing the update spectrum on the active subspace. This property concerns the update, not the conditioning of the model or objective. In practice, Muon replaces the SVD with five Newton–Schulz iterations: $O_t = \mathrm{NS}_5(M_t), W_t = W_{t-1} - \eta_t O_t$, where NS_5 includes input normalization and approximates the polar factor $U_t V_t^\top$ via matrix multiplications, whereas our analysis uses the exact LMO.

3.2 CHALLENGES OF MUON IN FL

Adapting Muon to heterogeneous federated learning presents two key challenges in aligning local momentum with the global gradient.

(Challenge 1) *Local LMO optimality does not guarantee global descent.*

Challenge Analysis: Starting from the same global model does not ensure consistent client directions. Each client computes $q_i^{r,k} = \mathrm{lmo}(M_i^{r,k+1})$ from its local momentum. For $v^r := \nabla f(x^r)$, the exact LMO satisfies $\langle v^r, q_i^{r,k}\rangle \le -\|v^r\|_\star + 2\|M_i^{r,k+1} - v^r\|_\star$. Thus, global descent depends on how closely local momentum tracks the global gradient. Our convergence analysis separates this mismatch into client momentum disagreement and the tracking error of the averaged momentum. Data heterogeneity and local trajectory drift contribute to the former, while matrix orthogonalization alone does not control either error. The challenge is therefore to coordinate local directions while maintaining an accurate global momentum reference.

(Challenge 2) *Momentum reinitialization.*

Challenge Analysis: In FL, `Local Muon` does not preserve the momentum state across communication rounds. Instead, at the beginning of each round, each client reinitializes its local momentum using only the current round information, e.g., $M_i^{r,0} \leftarrow \nabla F_i(\boldsymbol{x}_i^{r,0}; \xi_i^{r,0})$. As a result, the optimizer loses temporal information accumulated in previous rounds, which weakens momentum tracking and slows convergence. Moreover, because the momentum must be rebuilt from scratch on each client using only local data, the resulting local update directions become more sensitive to client-specific distribution shifts, thereby further aggravating client drift under non-IID data.

4 OUR ALGORITHM: FMUON

To robustly leverage matrix orthogonalization in FL, we propose `FMuon`, with two mechanisms.

Algorithm 1 FMuon Algorithm

Initialize global model $\boldsymbol{x}^0$, global direction $\Delta_G^0 = 0$, momentum $\bar{\boldsymbol{M}}^0 = \mathbf{0}$, β, the total number of clients N, and the number of selected clients per round $|S_r| = S$.
for $r = 0, \ldots, R-1$ **do**
  Server samples a client subset $\mathcal{S}_r \subset [N]$ uniformly at random with $|\mathcal{S}_r| = S$
  for each client $i \in S_r$ in parallel **do**
    $\boldsymbol{x}_i^{r,0} \leftarrow \boldsymbol{x}^r$, $\boldsymbol{M}_i^{r,0} \leftarrow \bar{\boldsymbol{M}}^r$
    for $k = 0, \ldots, K-1$ **do**
      $\boldsymbol{G}_i^{r,k} \leftarrow \nabla f_i(\boldsymbol{x}_i^{r,k}; \xi_i)$; $\quad \boldsymbol{M}_i^{r,k+1} \leftarrow \beta \boldsymbol{M}_i^{r,k} + (1-\beta)\boldsymbol{G}_i^{r,k}$
      $q_i^{r,k} \leftarrow \mathrm{lmo}(\boldsymbol{M}_i^{r,k+1})$; $\quad \boldsymbol{x}_i^{r,k+1} \leftarrow \boldsymbol{x}_i^{r,k} + \eta \left[(1-\alpha) q_i^{r,k} - \alpha \boldsymbol{\Delta}_G^r\right]$
    end for
    Communicate $(\boldsymbol{x}_i^{r,K} - \boldsymbol{x}_i^{r,0}, \boldsymbol{M}_i^{r,K})$ to the server
  end for
  $\boldsymbol{\Delta}_G^{r+1} \leftarrow -\frac{1}{SK\eta} \sum_{i \in \mathcal{S}_r} (\boldsymbol{x}_i^{r,K} - \boldsymbol{x}_i^{r,0})$; $\quad \boldsymbol{x}^{r+1} \leftarrow \boldsymbol{x}^r + \frac{1}{S} \sum_{i \in \mathcal{S}_r} (\boldsymbol{x}_i^{r,K} - \boldsymbol{x}_i^{r,0})$
  $\bar{\boldsymbol{M}}^{r+1} \leftarrow \frac{1}{S} \sum_{i \in \mathcal{S}_r} \boldsymbol{M}_i^{r,K}$
end for

4.1 Mechanism I: Local–Global Alignment

(Q1) *How to overcome Local LMO optimality does not guarantee global descent?*

To address **Challenge 1**, we incorporate local-global alignment into the local update rule:

$$q_i^{r,k} \leftarrow \mathrm{lmo}(\boldsymbol{M}_i^{r,k+1}), \boldsymbol{x}_i^{r,k+1} \leftarrow \boldsymbol{x}_i^{r,k} + \eta \left[(1-\alpha) q_i^{r,k} - \alpha \boldsymbol{\Delta}_G^r\right], \tag{3}$$

where $\boldsymbol{\Delta}_G^r = -\frac{1}{SK\eta} \sum_{i \in \mathcal{S}_r} (\boldsymbol{x}_i^{r-1,K} - \boldsymbol{x}_i^{r-1,0})$ is the estimated global update. α is the trade-off coefficient between local and global updates. We also validate its effectiveness in the following experiments (see Table 5 below).

4.2 Mechanism II: Momentum Aggregation

(Q2) *How to initialize momentum of Muon in local?*

To achieve better initialization of the momentum $\boldsymbol{M}$ in local, we aggregate local momentum $\boldsymbol{M}_i^{r,K}$ and transmit the aggregated result $\bar{\boldsymbol{M}}$ back to the clients. This strategy partially mitigates the client drift caused by reinitializing momentum from zero, and better aligns local updates with the global update direction (see **Table 5** below).

(Q3) *How to efficiently communicate momentum?*

Momentum Compression via SVD. Directly communicating the full momentum matrix M would introduce prohibitive communication overhead. To reduce the cost, we compress M using singular value decomposition (SVD): $M = U\Sigma V^\top$, where U and V are orthogonal matrices and Σ is the diagonal matrix of singular values. Instead of transmitting the full decomposition, we retain only the top-k singular values (with k set to 2.5% of the matrix rank), yielding a low-rank approximation (see Figure 2): $M \approx U_k \Sigma_k V_k^\top$. We communicate $(U_k, \Sigma_k, V_k^\top)$ instead of M. This significantly reduces the communication cost 95%. We refer to this variant as FMuon_SVD. In the following experiments, we show that this approach achieves performance comparable to FMuon (Table 4).

5 Theoretical Analysis

We establish convergence of FMuon and a strict separation from Local Muon on a heterogeneous problem family. Let $f(x) = N^{-1} \sum_{i=1}^N f_i(x)$, assume $f^\star := \inf_x f(x) > -\infty$, with $\Delta = f(x^0) - f^\star$. We use a primal norm $\|\cdot\|$, its dual $\|\cdot\|_\star$, and $\rho := \sup_{v \neq 0} \|v\|_\star / \|v\|_F$. We use the dual-gradient norm $G(x) := \|\nabla f(x)\|_\star = \max_{\|q\| \leq 1} \langle \nabla f(x), -q \rangle, \mathcal{G}_R := \frac{1}{R} \sum_{r=0}^{R-1} \mathbb{E}[G(x^r)]$. For matrix parameters with spectral primal norm, the dual norm is the nuclear norm.

Assumption 5.1 (Smoothness)**.** For every $i \in [N]$ and all x, y, $\|\nabla f_i(x) - \nabla f_i(y)\|_\star \le L\|x - y\|$.

Assumption 5.2 (Stochastic gradient oracle)**.** Let $\mathcal{F}_{r,k}$ denote the history immediately before the gradient queries at local step k of round r, including the current local iterates. For $e_i^{r,k} := g_i^{r,k} - \nabla f_i(x_i^{r,k})$, $\mathbb{E}[e_i^{r,k} \mid \mathcal{F}_{r,k}] = 0, \mathbb{E}[\|e_i^{r,k}\|_F^2 \mid \mathcal{F}_{r,k}] \le \sigma^2$. For distinct clients $i \ne j$, the simultaneous oracle errors satisfy $\mathbb{E}[\langle e_i^{r,k}, e_j^{r,k}\rangle_F \mid \mathcal{F}_{r,k}] = 0$.

Assumption 5.3 (Bounded heterogeneity)**.** For every x, $\frac{1}{N}\sum_{i=1}^N \|\nabla f_i(x) - \nabla f(x)\|_F^2 \le \zeta^2$.

Theorem 5.4 (Convergence of Local Muon)**.** *Suppose Assumptions 5.1, 5.2, and 5.3 hold, and $f^\star := \inf_x f(x) > -\infty$. Consider full participation with exact LMOs. Let $\delta := 1 - \beta \in (0, 1]$, $\Delta := f(x^0) - f^\star$, and $G(x) := \|\nabla f(x)\|_\star$. For any constant stepsize $\eta > 0$,*

$$\frac{1}{R}\sum_{r=0}^{R-1} \mathbb{E}[G(x^r)] \le \frac{\Delta}{\eta K R} + \frac{L\eta K}{2} + L\eta(K-1) + \frac{2\beta L\eta}{\delta} + 2\rho\zeta + \frac{2\rho\sigma}{\delta K} + 2\rho\sigma\sqrt{\delta}. \tag{4}$$

In particular, choose $\eta = \frac{\eta_0}{K\sqrt{R}}, 1 - \beta = K^{-2/3}$, where $\eta_0 > 0$ is independent of R. Then

$$\frac{1}{R}\sum_{r=0}^{R-1} \mathbb{E}[G(x^r)] \le \left[\frac{\Delta}{\eta_0} + L\eta_0\left(\frac{1}{2} + \frac{K-1}{K} + \frac{2\beta}{K^{1/3}}\right)\right] R^{-1/2} + \boxed{2\rho\zeta + \frac{4\rho\sigma}{K^{1/3}}}. \tag{5}$$

For fixed K, the bound consists of an $O(R^{-1/2})$ transient term and an $O(\rho\zeta + \rho\sigma K^{-1/3})$ residual.

Theorem 5.5 (Convergence of `FMuon`)**.** *Consider full participation with exact LMOs. Under Assumptions 5.1–5.3, fix K, N, α and $\eta_0 > 0$ independently of R, and choose $\eta = \frac{\eta_0}{KR^{3/4}}, 1 - \beta = R^{-1/2}$.*

$$\mathcal{G}_R \le \left(\frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \frac{2\rho\sigma}{\sqrt{N}}\right) R^{-1/4} + \left(\frac{2E_0}{K} + \rho(K+1)(2\sigma + \zeta)\right) R^{-1/2} + C_\alpha R^{-3/4} + C_K R^{-5/4},$$

where $C_\alpha = L\eta_0\left(\frac{1}{2} + \frac{\alpha}{1-\alpha} + \frac{2(K-1)}{K}\right), C_K = \frac{2L\eta_0(K+1)(K-1)}{K}$. Consequently, $\mathcal{G}_R = O(R^{-1/4})$.

Removing uniform heterogeneity. Under full participation, Assumption 5.3 is not necessary for the $O(R^{-1/4})$ rate. Let $J_0 = N^{-1}\sum_i \|\nabla f_i(x^0) - \nabla f(x^0)\|_\star$. Under only Assumptions 5.1 and 5.2, the stochastic schedule above gives

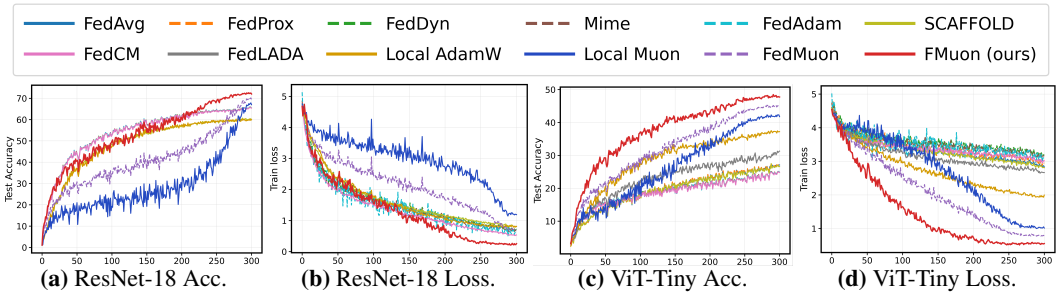
$$\mathcal{G}_R \le \left(\frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \frac{2\rho\sigma}{\sqrt{N}} + 2L\eta_0(K+1)\right) R^{-1/4} + \left(\frac{2E_0}{K} + (K+1)(2\rho\sigma + J_0)\right) R^{-1/2} + C_\alpha R^{-3/4} + C_K R^{-5/4}. \tag{6}$$

This follows from the bounded global displacement $\|x^r - x^0\| \le r\eta K$ and smoothness.

Interpretation and comparison. The two bounds distinguish transient decay from residual error. For fixed K, Local Muon has an $O(R^{-1/2})$ transient term, but its guarantee retains an $O(\rho\zeta + \rho\sigma K^{-1/3})$ residual. In contrast, `FMuon` guarantees $\mathcal{G}_R = O(R^{-1/4})$, with all terms vanishing: heterogeneity contributes only through an $O(R^{-1/2})$ term, while the leading stochastic term scales as $\sigma/\sqrt{N}$. Thus, the theoretical advantage is convergence to stationarity without a persistent residual in the bound, rather than a faster transient exponent. The Local Muon upper bound alone does not establish an unavoidable error floor. Under full participation, uniform bounded heterogeneity is unnecessary: client-wise smoothness and the algorithm-induced displacement bound provide trajectory-dependent control of gradient disagreement, preserving the $O(R^{-1/4})$ rate without assuming bounded gradients or a bounded parameter domain. Appendix E analyzes `FMuon` with partial client participation.

6 Experiments

Datasets. We evaluate `FMuon` on both vision and language tasks. (*i*) For image classification, we use CIFAR-100 (Krizhevsky et al., 2009), and Tiny ImageNet (Le & Yang, 2015). (*ii*) For NLP tasks, we adopt benchmark datasets from the GLUE benchmark (Wang et al., 2019), including SST-2 (Wang et al., 2019), QQP (Wang et al., 2019), and OpenWebText Gokaslan & Cohen (2019) dataset. To simulate data heterogeneity across clients, we follow the Dirichlet partitioning scheme (Hsu et al., 2019), where a Dir-0.6 corresponds to a low heterogeneity and Dir-0.1 implies high heterogeneity.



(a) ResNet-18 Acc. **(b)** ResNet-18 Loss. **(c)** ViT-Tiny Acc. **(d)** ViT-Tiny Loss.

Figure 5: Training loss and test accuracy curves on CIFAR-100 with ResNet-18 and ViT-Tiny under the Dir-0.1.

Table 1: Test accuracy (%) and training loss on CIFAR-100 with ResNet-18 and ViT-Tiny after 300 communication rounds under Dir-0.6 and Dir-0.1. All experiments use 100 clients, 10% participation, a batch size of 50, and $K = 50$ local steps. Best and second-best results are **bold** and underlined.

Method	ResNet-18 (Dir-0.6)		ResNet-18 (Dir-0.1)		ViT-Tiny (Dir-0.6)		ViT-Tiny (Dir-0.1)	
	Acc. ↑	Loss ↓	Acc. ↑	Loss ↓	Acc. ↑	Loss ↓	Acc. ↑	Loss ↓
FedAvg	64.08 ± 0.18	0.376	60.25 ± 0.20	0.767	32.36 ± 0.08	2.350	27.14 ± 0.12	2.867
FedProx	63.12 ± 0.15	0.458	59.66 ± 0.28	0.812	31.51 ± 0.12	2.425	26.84 ± 0.15	2.875
FedDyn	66.12 ± 0.28	0.352	63.01 ± 0.28	0.615	33.25 ± 0.22	2.125	27.66 ± 0.18	2.723
Mime	67.34 ± 0.21	0.312	63.37 ± 0.18	0.604	34.12 ± 0.14	2.101	27.76 ± 0.22	2.702
FedAdam	67.23 ± 0.18	0.332	63.61 ± 0.21	0.512	34.32 ± 0.32	1.965	28.50 ± 0.11	2.425
SCAFFOLD	65.01 ± 0.19	0.365	62.56 ± 0.16	0.814	32.17 ± 0.12	2.295	27.31 ± 0.18	2.855
FedCM	68.42 ± 0.11	0.282	65.73 ± 0.14	0.639	33.33 ± 0.17	2.081	31.18 ± 0.13	2.338
FedLADA	65.07 ± 0.17	0.671	57.78 ± 0.18	0.498	38.33 ± 0.12	2.121	31.50 ± 0.19	2.678
Local AdamW	62.84 ± 0.08	0.363	58.97 ± 0.10	0.794	40.47 ± 0.09	1.026	37.86 ± 0.11	1.954
FedMuon	72.23 ± 0.21	0.175	69.71 ± 0.23	0.812	48.09 ± 0.25	0.181	45.17 ± 0.27	0.862
`Local Muon`	71.66 ± 0.15	0.395	66.71 ± 0.15	1.504	46.69 ± 0.15	0.201	40.53 ± 0.17	1.432
`FMuon (ours)`	$\mathbf{74.12 \pm 0.18}$	**0.021**	$\mathbf{73.05 \pm 0.16}$	**0.246**	$\mathbf{50.22 \pm 0.14}$	**0.162**	$\mathbf{48.18 \pm 0.12}$	**0.556**

Model Architectures. We explore a variety of model types: (*i*) ResNet-18 (He et al., 2016) as a representative convolutional neural network (CNN), (*ii*) Swin Transformer (Liu et al., 2021) and ViT-Tiny (Dosovitskiy et al., 2020) for Vision Transformers, and (*iii*) RoBERTa-Base (Liu et al., 2019) and GPT-2 Radford et al. (2019) for large-scale language model.

Baselines. We compare our method against state-of-the-art FL algorithms: `FedAvg` (`Local SGD`) (McMahan et al., 2017), `SCAFFOLD` (Karimireddy et al., 2020b), `FedCM` (Xu et al., 2021), `FedLADA` (Sun et al., 2023), `Local AdamW` and `Local Muon`, `FedProx` Li et al. (2020b), `FedDyn` Durmus et al. (2021), `Mime` Karimireddy et al. (2020a), and `FedAdam` Reddi et al.. In the Appendix (Table 14), we compare additional FL algorithms designed to address data heterogeneity.

Hyperparameter Settings. For `FedAvg`, `SCAFFOLD`, `FedCM`, `FedProx`, `FedDyn`, `Mime`, and `FedAdam`, the lr is selected from $\{10^{-2},\ 3\times10^{-2},\ 5\times10^{-2},\ 10^{-1},\ 3\times10^{-1}\}$, with a weight decay of 0.001. For `FedLADA`, `Local AdamW`, the lr is selected from $\{10^{-4},\ 3\times10^{-4},\ 5\times10^{-4},\ 8\times10^{-4},\ 10^{-3}\}$, with weight decay 0.01 or 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$. We apply cosine learning rate decay, and set `FMuon` to $\boldsymbol{\alpha} = 0.5$, weight decay 0.01, $\beta = 0.95$ following the official default setting. We set the learning rate of `FMuon` and `Local Muon` to be selected from $\{3\times10^{-2},\ 2\times10^{-2},\ 3\times10^{-3}, 1\times10^{-3}\}$. Additional hyperparameter configurations are detailed in the Appendix (see Table 8 and Table 11 in Appendix). We release all code, configuration files to ensure full reproducibility. All results are averaged over 5 runs with std reported with seeds 42, 43, 44, 45, 46. All experiments were performed on NVIDIA RTX 4090 GPU.

6.1 Results on CNN and Transformer

Training on CIFAR-100 with ResNet-18. Table 1 and Figure 5 present the test accuracy and training loss on CIFAR-100 using ResNet-18. `FMuon` achieves the best performance under both Dir-0.6 and Dir-0.1 settings, reaching a top accuracy of 74.12% and 73.05%, respectively. It also attains the lowest training loss (0.021 and 0.246), demonstrating faster and more stable convergence. Compared to other adaptive baselines such as `Local AdamW`, `FMuon` shows superior generalization under data heterogeneity, confirming its effectiveness in CNNs. In our experiments, Muon provides

Table 2: Test accuracy (%) of RoBERTa-Base with LoRA on GLUE tasks over 100 communication rounds, Dir-0.5. Experiments are conducted with 20 clients, 20% participation, batch size 16, and $K = 50$.

Method	GLUE Tasks							Avg.
	CoLA	RTE	SST-2	QQP	MRPC	QNLI	MNLI	
FedAvg	51.00 ± 0.26	51.99 ± 0.24	93.04 ± 0.16	81.75 ± 0.11	88.24 ± 0.18	89.36 ± 0.15	81.72 ± 0.25	76.73
FedProx	53.11 ± 0.14	53.25 ± 0.21	92.26 ± 0.18	81.15 ± 0.11	87.36 ± 0.12	88.12 ± 0.14	81.41 ± 0.21	76.67
FedDyn	53.21 ± 0.28	52.22 ± 0.30	92.36 ± 0.21	81.35 ± 0.21	87.89 ± 0.11	89.12 ± 0.21	82.18 ± 0.21	76.90
Mime	52.15 ± 0.17	51.62 ± 0.21	92.21 ± 0.28	80.26 ± 0.18	88.04 ± 0.12	89.11 ± 0.21	82.51 ± 0.20	76.56
FedAdam	53.21 ± 0.28	52.52 ± 0.31	92.36 ± 0.25	82.22 ± 0.28	88.12 ± 0.34	88.01 ± 0.23	82.66 ± 0.22	77.01
SCAFFOLD	52.15 ± 0.17	50.65 ± 0.20	93.28 ± 0.28	80.26 ± 0.18	88.35 ± 0.12	89.32 ± 0.24	82.11 ± 0.20	76.59
FedCM	53.21 ± 0.28	52.22 ± 0.30	92.56 ± 0.25	81.22 ± 0.28	88.56 ± 0.13	89.02 ± 0.23	82.12 ± 0.27	76.99
FedLADA	54.66 ± 0.17	57.02 ± 0.08	$\underline{93.88 \pm 0.16}$	81.56 ± 0.20	$\underline{89.01 \pm 0.28}$	89.86 ± 0.29	82.44 ± 0.17	78.35
Local AdamW	55.38 ± 0.12	59.57 ± 0.25	93.81 ± 0.19	81.51 ± 0.05	88.73 ± 0.23	89.55 ± 0.15	82.86 ± 0.26	78.77
`Local Muon`	$\underline{55.54 \pm 0.05}$	$\underline{64.93 \pm 0.17}$	93.58 ± 0.27	$\underline{83.06 \pm 0.11}$	88.95 ± 0.13	$\underline{90.52 \pm 0.27}$	$\underline{84.63 \pm 0.10}$	$\underline{80.17}$
`FMuon (ours)`	$\mathbf{56.78 \pm 0.11}$	$\mathbf{66.58 \pm 0.29}$	$\mathbf{94.04 \pm 0.25}$	$\mathbf{84.65 \pm 0.16}$	$\mathbf{89.11 \pm 0.07}$	$\mathbf{90.64 \pm 0.13}$	$\mathbf{85.21 \pm 0.18}$	**81.00**

Table 3: Test accuracy (%) on CIFAR-100 under Dir-0.1 and validation loss on OpenWebText across model sizes after 300 communication rounds. CIFAR-100 uses 100 clients, 10% participation, batch size 50, and $K = 50$ local steps; OpenWebText uses 20 clients, 20% participation, batch size 16, and $K = 100$ local steps. Best and second-best results are **bold** and underlined, respectively.

Method	CIFAR-100 (Acc. ↑)				OpenWebText (Val. Loss ↓)			
	ViT-Tiny	ViT-Small	ViT-Base	ViT-Large	GPT-2 S	GPT-2 M	GPT-2 L	GPT-2 XL
FedAvg	27.14	29.52	31.15	33.56	4.25	4.12	4.01	3.91
FedProx	26.84	28.63	31.05	33.25	4.33	4.21	4.15	4.05
FedDyn	27.66	30.56	32.91	35.28	4.08	4.01	3.88	3.81
Mime	27.76	31.23	33.11	35.34	4.10	4.02	3.89	3.78
FedAdam	28.50	33.15	34.56	36.13	4.02	3.95	3.82	3.75
SCAFFOLD	27.31	30.24	32.85	34.58	4.12	4.01	3.95	3.82
FedCM	31.18	25.15	27.88	29.01	4.32	4.21	4.02	3.91
FedLADA	31.50	33.15	36.21	38.10	3.56	3.45	3.33	3.24
Local AdamW	37.86	38.64	40.16	42.21	3.44	3.35	3.27	3.15
FedMuon	<u>45.17</u>	<u>47.17</u>	<u>49.45</u>	<u>52.67</u>	<u>3.28</u>	<u>3.16</u>	<u>3.04</u>	<u>2.89</u>
`Local Muon`	40.53	42.34	45.26	46.54	3.33	3.21	3.09	2.98
`FMuon (ours)`	**48.18**	**50.52**	**53.63**	**56.24**	**3.12**	**2.98**	**2.85**	**2.74**

immediate speedups under IID data, but under non-IID data Local Muon initially converges slowly due to mismatched client lmo and exacerbated client drift (Challenge 1). `FMuon` mitigates this issue, achieving fast and stable convergence, and also yields clear speedups in IID settings (Table 15).

Training on CIFAR-100 with ViT-Tiny. Table 1 and Figure 5 show that `FMuon` achieves the best performance across both heterogeneity levels, with 50.22% (Dir-0.6) and 48.18% (Dir-0.1), and the lowest training loss (0.162 and 0.556), confirming its efficient convergence. These results validate that `FMuon` is particularly effective for federated vision Transformers under non-IID conditions. The small dataset CIFAR100 is difficult to support the performance of ViT, resulting in lower accuracy. Therefore, we continued to test on the pretrained model. FedCM performs strongly on ResNet-18 but

Table 4: Ablation studies of `FMuon` on CIFAR-100 under Dir-0.1 partition.**Comm.** denotes communication cost per round (MB), and **Comp.** denotes computation time per round (s).

Sensitivity to α						Momentum Aggregation			
Model	0.00	0.25	**0.50**	0.75	0.90	**Method**	**Acc. (%) ↑**	**Comm. ↓**	**Comp. ↓**
ResNet-18	68.05	69.89	**73.05**	72.12	67.56	NoAgg	69.12	46.8	6.23
ViT-Tiny	44.56	46.28	**48.18**	47.59	46.23	Agg-$\boldsymbol{m}$	73.05	93.6	6.44
GPT-2 S	3.19	3.16	**3.12**	3.15	3.18	Agg-$\boldsymbol{m}$-SVD	72.56	49.2	6.48

Table 5: Ablation study of FMuon on CIFAR-100 under the Dir-0.1 partition after 300 communication rounds. Results are reported on **ResNet-18** and **ViT-Tiny**. Best results in each block are highlighted in **bold**.

Effect of Removing Components			**Effect of Different Local Optimizers**		
Variant	**ResNet-18**	**ViT-Tiny**	**Variant**	**ResNet-18**	**ViT-Tiny**
A1: w/o $\bar{\boldsymbol{m}}$	69.12 ± 0.18	43.67 ± 0.19	Local SGD + $\bar{\boldsymbol{m}}$ + $\boldsymbol{\Delta}_G$	66.28 ± 0.17	32.56 ± 0.11
A2: w/o $\boldsymbol{\Delta}_G$	68.05 ± 0.10	44.56 ± 0.16	Local AdamW + $\bar{\boldsymbol{m}}$ + $\boldsymbol{\Delta}_G$	64.25 ± 0.12	41.26 ± 0.17
A3: FMuon	$\mathbf{73.05 \pm 0.15}$	$\mathbf{48.18 \pm 0.12}$	Local Muon + $\bar{\boldsymbol{m}}$ + $\boldsymbol{\Delta}_G$	$\mathbf{73.05 \pm 0.15}$	$\mathbf{48.18 \pm 0.12}$

poorly on ViT-Tiny mainly because it relies on SGD-style local optimization. While such updates are often adequate for CNNs, they tend to optimize Transformer models much more slowly, making FedCM less suitable for ViT-Tiny in our setting.

Fine-tuning Results on LLMs. Table 2 summarizes the results on the GLUE benchmark using RoBERTa-Base with LoRA, 20 clients, 20% participation, batch size 16, $K = 50$, rank=16. FMuon achieves the highest accuracy of GLUE outperforming strong baselines such as FedAvg and Local Muon. It is particularly strong on challenging tasks like RTE and QQP, exceeding the next best methods by +1.65% and +1.59%, respectively.

Pre-training Results on Transformers. Table 3 shows that FMuon achieves the highest CIFAR-100 accuracy and lowest OpenWebText validation loss across all evaluated model sizes. Under Dir-0.1, it improves ViT accuracy over Local Muon by 7.65–9.70 percentage points and reduces GPT-2 validation loss by 0.21–0.24, demonstrating consistent benefits across model scales and both vision and language tasks.

6.2 Ablation Study

Impact of Δ_G and $\bar{m}$. As shown in Table 5, we conduct an ablation study of FMuon. FMuon incorporates momentum averaging $\bar{m}$ and global update differences Δ_G. The results clearly indicate that Local Muon consistently outperforms both SGD and AdamW, demonstrating its superior ability to handle non-IID FL. Our strategy consistently improves the performance of other optimizers.

Impact of Δ_G and $\bar{m}$ on other optimizers. As shown in Table 5, we compare different local optimizers with Δ_G and $\bar{m}$. The results demonstrate that Local Muon consistently achieves the best performance, significantly outperforming SGD and AdamW, thereby highlighting its effectiveness in mitigating data heterogeneity. Further more results on additional optimizers in Table 19 in Appendix.

Acceleration of Matrix Orthogonalization on Federated Learning. See Table 18 in Appendix. Matrix orthogonalization also provides acceleration for other FL algorithms.

Impact of α. Table 4 evaluates the effect of the local-global alignment parameter α in FMuon. We observe that $\alpha = 0.5$ yields the best performance, striking a balance between local adaptivity and global consistency.

Impact of Momentum Aggregation Strategy. Table 4 shows Momentum Aggregation Strategy, Agg-$\boldsymbol{m}$-SVD (FMuon_SVD), achieves the best balance between accuracy and communication cost. While Agg-$\boldsymbol{m}$ improves performance, it introduces excessive communication ($2\times$). In contrast, Agg-$\boldsymbol{m}$-SVD attains similar benefits with only $1.05\times$ communication cost.

7 Conclusion

We proposed FMuon, a structure-aware federated optimizer that combines matrix-orthogonalized updates, local-global alignment, cross-round momentum aggregation, and low-rank state sharing to mitigate client drift, stabilize optimization, and reduce communication under non-IID data. We established non-convex convergence guarantees and demonstrated strong empirical gains across vision and language tasks, especially for Transformers. These findings support matrix-aware optimization for efficient federated learning and motivate extensions to optimizers such as LAMB (You et al., 2020) and SOAP (Vyas et al., 2025), with potential applications to distributed training and parameter-efficient fine-tuning.

REFERENCES

Jeremy Bernstein and Laker Newhouse. Modular duality in deep learning. *arXiv preprint arXiv:2410.21265*, 2024.

Ronan Collobert. Large scale machine learning. *Idiap Res. Inst., Martigny, Switzerland, RR-04-42*, 2004.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Alp Emre Durmus, Zhao Yue, Matas Ramon, Mattina Matthew, Whatmough Paul, and Saligrama Venkatesh. Federated learning based on dynamic regularization. In *International conference on learning representations*, 2021.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. https://skylion007.github.io/OpenWebTextCorpus, 2019.

Vineet Gupta, Tomer Koren, and Yoram Singer. Shampoo: Preconditioned stochastic tensor optimization. In *International Conference on Machine Learning*, pp. 1842–1850. PMLR, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

Keller Jordan, Yuchen Jin, Vlado Boza, You Jiacheng, Franz Cecista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. *URL https://kellerjordan. github. io/posts/muon*, 6, 2024.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020a.

Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pp. 5132–5143. PMLR, 2020b.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, 2020a.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020b.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, et al. Muon is scalable for llm training. *arXiv preprint arXiv:2502.16982*, 2025.

Junkang Liu, Fanhua Shang, Yuanyuan Liu, Hongying Liu, Yuangang Li, and YunXiang Gong. Fedbcgd: Communication-efficient accelerated block coordinate gradient descent for federated learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 2955–2963, 2024.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained lmos. In *International Conference on Machine Learning*, pp. 49069–49104. PMLR, 2025.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*.

Ishaan Shah, Anthony M Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, Darsh J Shah, et al. Practical efficiency of muon for pretraining. *arXiv preprint arXiv:2505.02222*, 2025.

Yan Sun, Li Shen, Hao Sun, Liang Ding, and Dacheng Tao. Efficient federated learning via local adaptive amended optimizer with linear speedup. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):14453–14464, 2023.

Yuki Takezawa, Anastasia Koloskova, Xiaowen Jiang, and Sebastian Stich. Fedmuon: Federated learning with bias-corrected lmo-based optimization. In *International Conference on Learning Representations*, volume 2026, pp. 39538–39569, 2026.

Benjamin Thérien, Xiaolong Huang, Irina Rish, and Eugene Belilovsky. Muloco: Muon is a practical inner optimizer for diloco. In *ES-FoMo III: 3rd Workshop on Efficient Systems for Foundation Models*, 2025.

Nikhil Vyas, Depen Morwani, Rosie Zhao, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham M. Kakade. Soap: Improving and stabilizing shampoo using adam for language modeling. In *International Conference on Learning Representations*, 2025. URL https://api.semanticscholar.org/CorpusID:278498822.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*, 2019.

Jing Xu, Sen Wang, Liwei Wang, and Andrew Chi-Chih Yao. Fedcm: Federated learning with client-level momentum. *arXiv preprint arXiv:2106.10874*, 2021.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. In *International Conference on Learning Representations (ICLR)*, 2020.

Yushun Zhang, Congliang Chen, Ziniu Li, Tian Ding, Chenwei Wu, Diederik Durk Kingma, Yinyu Ye, Zhi-Quan Luo, and Ruoyu Sun. Adam-mini: Use fewer learning rates to gain more. In *International Conference on Learning Representations*, volume 2025, pp. 28033–28063, 2025.

ETHICS STATEMENT

This work studies federated optimization for vision and language models using existing benchmarks, including CIFAR-100, Tiny ImageNet, OpenWebText, and GLUE tasks. Our experiments simulate federated clients using benchmark partitions rather than collecting data from real users. Although FMuon keeps raw training data local, communicated model updates and momentum information may reveal information about local data. Neither federated training nor momentum orthogonalization alone provides a formal privacy guarantee; FMuon does not establish differential privacy or resistance to inference attacks. Models trained with this approach may also inherit biases or harmful content from the underlying datasets and pretrained models. Deployment with sensitive data therefore requires additional privacy safeguards and application-specific safety evaluation.

REPRODUCIBILITY STATEMENT

The paper presents the FMuon algorithm, including momentum aggregation and local–global alignment, and states the assumptions underlying its convergence guarantees. Detailed derivations and proofs are provided in the appendix. The main text and appendix describe the datasets, model architectures, client partitioning and participation settings, communication rounds, local training configurations, optimizer hyperparameters, and learning-rate selection. Results are averaged over five runs using seeds 42, 43, 44, 45, and 46, with standard deviations reported. Experiments were conducted on NVIDIA RTX 4090 GPUs. The implementation and configuration files are available through the anonymous repository linked in the paper.

AI USE STATEMENT

We used large language models to assist with language editing, including sentence rephrasing, grammar correction, and readability improvements. We did not use these tools for research ideation, methodology or experiment design, theoretical derivations, method implementation, data generation or processing, or interpretation of results. The authors reviewed and revised all AI-assisted text and take full responsibility for the final manuscript, including its claims, analyses, and conclusions.

ETHICS STATEMENT

This work studies federated optimization for vision and language models using existing benchmarks, including CIFAR-100, Tiny ImageNet, OpenWebText, and GLUE tasks. Our experiments simulate federated clients using benchmark partitions rather than collecting data from real users. Although FMuon keeps raw training data local, communicated model updates and momentum information may reveal information about local data. Neither federated training nor momentum orthogonalization alone provides a formal privacy guarantee; FMuon does not establish differential privacy or resistance to inference attacks. Models trained with this approach may also inherit biases or harmful content from the underlying datasets and pretrained models. Deployment with sensitive data therefore requires additional privacy safeguards and application-specific safety evaluation.

REPRODUCIBILITY STATEMENT

The paper presents the FMuon algorithm, including momentum aggregation and local–global alignment, and states the assumptions underlying its convergence guarantees. Detailed derivations and proofs are provided in the appendix. The main text and appendix describe the datasets, model architectures, client partitioning and participation settings, communication rounds, local training configurations, optimizer hyperparameters, and learning-rate selection. Results are averaged over five runs using seeds 42, 43, 44, 45, and 46, with standard deviations reported. Experiments were conducted on NVIDIA RTX 4090 GPUs. The implementation and configuration files are available through the anonymous repository linked in the paper.

AI USE STATEMENT

We used large language models to assist with language editing, including sentence rephrasing, grammar correction, and readability improvements. We also used these tools to assist with appendix organization and manuscript consistency checks, including checking arithmetic in reported comparisons and identifying mismatches between tables and descriptions. The authors reviewed and revised all AI-assisted text and take full responsibility for the final manuscript, including its claims, analyses, and conclusions.

APPENDIX CONTENTS

Theoretical Analysis and Proofs 15

G: Experimental Setup 27

- Vision Datasets
- Federated Learning Configuration
- ResNet-18 Architecture
- ViT-Tiny Settings
- Vision Transformer Fine-Tuning Settings
- RoBERTa-Base Fine-Tuning Settings
- Federated Language-Model Settings

H: Additional Experiments 32

- Communication and Computation Costs
- Additional Non-IID Baselines
- IID and Non-IID Comparisons
- Language-Model Fine-Tuning Results
- Results across Model Scales
- Muon across Federated Algorithms
- Relation to Existing Methods and Communication Accounting
- Correction across Local Optimizers

A ASSUMPTIONS AND COMMON NOTATION

The main comparison uses full participation and uniform client weighting. A separate extension below covers uniform partial participation. The parameter space is a finite-dimensional real inner-product space, equipped with a primal norm $\|\cdot\|$, its dual norm $\|\cdot\|_\star$, and the Euclidean/Frobenius norm $\|\cdot\|_F$. Let

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x), \qquad f^\star = \inf_x f(x) > -\infty, \qquad \Delta = f(x^0) - f^\star, \qquad \|v\|_\star \le \rho\|v\|_F.$$

The initial point x^0 is deterministic and $\rho < \infty$ is a norm-comparison constant. For a matrix with the spectral primal norm, the dual norm is the nuclear norm and one may take $\rho = \sqrt{\min\{m,n\}}$.

Assumption A.1 (Primal–dual smoothness)**.** Every f_i is continuously differentiable and

$$\|\nabla f_i(x) - \nabla f_i(y)\|_\star \le L\|x-y\| \quad \text{for all } x, y.$$

Consequently,

$$f(y) \le f(x) + \langle \nabla f(x), y - x\rangle + \frac{L}{2}\|y-x\|^2.$$

Assumption A.2 (Conditional stochastic oracle)**.** Let $\mathcal{F}_{r,k}$ contain all randomness available immediately before the local gradient queries at step (r,k), including the queried client iterates and all momentum states already constructed. Write

$$g_i^{r,k} = \nabla f_i(x_i^{r,k}) + \xi_i^{r,k}.$$

The oracle satisfies

$$\mathbb{E}[\xi_i^{r,k} \mid \mathcal{F}_{r,k}] = 0, \qquad \mathbb{E}[\|\xi_i^{r,k}\|_F^2 \mid \mathcal{F}_{r,k}] \le \sigma^2.$$

For $i \neq j$, the noises are conditionally uncorrelated:

$$\mathbb{E}[\langle \xi_i^{r,k}, \xi_j^{r,k}\rangle_F \mid \mathcal{F}_{r,k}] = 0.$$

Conditional independence across clients is sufficient. Independence of noise vectors across time is not required; the conditional zero-mean property makes them a martingale-difference sequence.

Assumption A.3 (Optional uniform gradient heterogeneity)**.** For all x,

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

Only results that explicitly invoke this assumption require it. Theorem C.8 dispenses with it under full participation.

We use the dual-gradient stationarity measure

$$G(x) = \|\nabla f(x)\|_\star = \max_{\|q\|\le 1}\langle \nabla f(x), -q\rangle.$$

This is a stationarity measure for the unconstrained problem, not a claim that the iterates are constrained to a unit ball. The exact LMO is a measurable selection

$$\mathrm{lmo}(m) \in \arg\min_{\|q\|\le 1}\langle m, q\rangle.$$

Lemma A.4 (Exact LMO perturbation inequality)**.** *If $q = \mathrm{lmo}(m)$, then $\|q\| \le 1$, $\langle m, q\rangle = -\|m\|_\star$, and for every v,*

$$\langle v, q\rangle \le -\|v\|_\star + 2\|v - m\|_\star.$$

Proof. The first two assertions follow from feasibility and duality. Moreover,

$$\langle v, q\rangle \le -\|m\|_\star + \|v-m\|_\star \le -\|v\|_\star + 2\|v-m\|_\star.$$

□

B LOCAL MUON: AN UPPER BOUND AND AN EXPLICIT FAILURE EXAMPLE

B.1 ALGORITHM AND UPPER BOUND

At the beginning of round r, Local Muon initializes $x_i^{r,0} = x^r$ and $m_i^{r,0} = g_i^{r,0}$, where $g_i^{r,0}$ is queried at x^r. For $k = 0, \dots, K-1$, it applies

$$q_i^{r,k} = \mathrm{lmo}(m_i^{r,k}), \qquad x_i^{r,k+1} = x_i^{r,k} + \eta q_i^{r,k}.$$

For $k < K-1$, it queries $g_i^{r,k+1}$ at $x_i^{r,k+1}$ and sets

$$m_i^{r,k+1} = \beta m_i^{r,k} + (1-\beta) g_i^{r,k+1}.$$

The server sets $x^{r+1} = N^{-1} \sum_i x_i^{r,K}$. Here $\eta > 0$, $\beta \in [0,1)$, and $K \geq 1$ are constant within a run.

Theorem B.1 (Local Muon upper bound)**.** *Under Assumptions A.1–A.3,*

$$\frac{1}{R}\sum_{r=0}^{R-1} \mathbb{E}G(x^r) \leq \frac{\Delta}{\eta K R} + \frac{L\eta K}{2} + L\eta(K-1) + 2\rho\zeta + \frac{2\rho\sigma}{(1-\beta)K} + \frac{2\beta L\eta}{1-\beta} + 2\rho\sigma\sqrt{1-\beta}. \tag{7}$$

Proof. The unit-norm directions give $\|x_i^{r,k} - x^r\| \leq k\eta$. Fix a round and a client, suppress these indices, and let $e_k = m_k - \nabla f_i(x_k)$ and $\xi_k = g_k - \nabla f_i(x_k)$. Since $e_0 = \xi_0$,

$$e_k = \beta^k \xi_0 + \beta \sum_{j=0}^{k-1} \beta^{k-1-j} (\nabla f_i(x_j) - \nabla f_i(x_{j+1})) + (1-\beta) \sum_{j=1}^{k} \beta^{k-j} \xi_j.$$

Smoothness bounds the second term in expected dual norm by $\beta L\eta/(1-\beta)$. For the last term, martingale orthogonality in Frobenius norm yields

$$\mathbb{E}\Big\|(1-\beta)\sum_{j=1}^{k} \beta^{k-j}\xi_j\Big\|_\star \leq \rho\sigma\sqrt{\frac{1-\beta}{1+\beta}} \leq \rho\sigma\sqrt{1-\beta}.$$

The initial term is at most $\beta^k \rho\sigma$. Averaging gives

$$\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}\|e_k\|_\star \leq \frac{\rho\sigma}{(1-\beta)K} + \frac{\beta L\eta}{1-\beta} + \rho\sigma\sqrt{1-\beta}.$$

Furthermore, smoothness, heterogeneity, and Jensen's inequality imply

$$\frac{1}{N}\sum_i \|\nabla f_i(x_i^{r,k}) - \nabla f(x^r)\|_\star \leq Lk\eta + \rho\zeta.$$

Writing $\bar{q}^{r,k} = N^{-1}\sum_i q_i^{r,k}$ and applying Lemma A.4,

$$\langle \nabla f(x^r), \bar{q}^{r,k} \rangle \leq -G(x^r) + 2Lk\eta + 2\rho\zeta + \frac{2}{N}\sum_i \|e_i^{r,k}\|_\star.$$

Because $x^{r+1} - x^r = \eta \sum_k \bar{q}^{r,k}$ has norm at most ηK, smoothness gives

$$\mathbb{E}f(x^{r+1}) \leq \mathbb{E}f(x^r) - \eta K \mathbb{E}G(x^r) + \frac{L}{2}\eta^2 K^2 + L\eta^2 K(K-1) + 2\eta K\rho\zeta + 2\eta K\left(\frac{\rho\sigma}{(1-\beta)K} + \frac{\beta L\eta}{1-\beta} + \rho\sigma\sqrt{1-\beta}\right).$$

Sum over rounds, use $f(x^R) \geq f^\star$, and divide by $\eta K R$. □

Corollary B.2 (One parameter choice for Local Muon). *For $\eta = \eta_0(KR)^{-1/2}$ and $1-\beta = K^{-2/3}$, the preceding bound has order*

$$O\left((KR)^{-1/2} + \sqrt{K/R} + K^{1/6}R^{-1/2} + \rho\zeta + \rho\sigma K^{-1/3}\right),$$

with L, Δ, η_0 absorbed in the constants.

Proof. Substitute the stated parameters into equation 7. □

Remark B.3 (What an upper bound does not establish). The term $2\rho\zeta$ in equation 7 is a residual term in an upper bound. It does not establish a lower bound on the actual stationarity error or nonconvergence on every heterogeneous problem. The next proposition supplies a separate existence result.

B.2 A MATRIX-VALUED FAILURE FAMILY, INCLUDING RETAINED LOCAL MOMENTUM

The noncommutativity obstruction is already studied by Takezawa et al., *FedMuon: Federated Learning with Bias-corrected LMO-based Optimization* (ICLR 2026, Theorem 1 and Appendix D). The following elementary family is provided to establish the precise comparison needed here: a spectral-norm matrix LMO, arbitrary fixed K, and both reset and retained local momentum. The underlying obstruction is not claimed as a new discovery.

Proposition B.4 (Nonstationary stagnation of Local Muon). *Let $W \in \mathbb{R}^{d\times d}$, use the spectral primal norm and nuclear dual norm, and consider two clients with*

$$f_1(W) = \frac{1}{2}\|W + 3cI_d\|_F^2, \qquad f_2(W) = \frac{1}{2}\|W - cI_d\|_F^2, \qquad c > 0.$$

Initialize $W^0 = 0$. For every $K \geq 1$, $0 < \eta K < c$, and $\beta \in [0, 1)$, each of the following deterministic Local Muon variants satisfies

$$W^r = 0, \qquad G(W^r) = dc \quad \textit{for all rounds } r:$$

1. *the reset-to-first-gradient variant analyzed in Theorem B.1;*
2. *the retained-momentum variant, initialized by $m_i^{0,0} = 0$, which queries a gradient, sets $m_i^{r,k+1} = \beta m_i^{r,k} + (1-\beta)\nabla f_i(W_i^{r,k})$, then applies $\mathrm{lmo}(m_i^{r,k+1})$, and retains $m_i^{r+1,0} = m_i^{r,K}$ across rounds.*

Both variants reset the local model to W^r at every round and average the K-step local endpoints. The assumptions hold with

$$L = d, \quad \rho = \sqrt{d}, \quad \sigma = 0, \quad \zeta = 2c\sqrt{d}, \quad \Delta = dc^2/2.$$

Proof. The global objective is

$$f(W) = \tfrac{1}{2}\|W + cI_d\|_F^2 + 2dc^2, \qquad \nabla f(W) = W + cI_d.$$

The two gradient deviations are $2cI_d$ and $-2cI_d$, giving the stated heterogeneity and initial gap. Since $\|A\|_* \leq d\|A\|_{\mathrm{sp}}$, the primal–dual smoothness constant $L = d$ is valid, and $\|A\|_* \leq \sqrt{d}\|A\|_F$ gives ρ.

Suppose $W^r = 0$. Induction over local steps gives $W_1^{r,k} = -k\eta I_d$ and $W_2^{r,k} = k\eta I_d$. The queried gradients are $(3c - k\eta)I_d$ and $(k\eta - c)I_d$, respectively, with strictly positive and strictly negative coefficients. For reset momentum, convex averaging preserves these respective signs. For retained momentum, the first gradient update creates the same signs from zero, and every subsequent update, including across rounds, preserves them. A nonzero scalar multiple aI_d has the unique spectral-ball LMO $-\mathrm{sign}(a)I_d$. Thus the local directions are $-I_d$ and I_d throughout the round, and the endpoints average to zero. Induction over rounds proves the claim. Finally, $G(0) = \|cI_d\|_* = dc$. All LMO inputs used in this argument are full rank; the conclusion does not depend on a choice of LMO at a singular input. □

C FMUON: CONVERGENCE WITH HISTORY-DIRECTION MIXING

C.1 ALGORITHM AND SCOPE

Fix $R, K \geq 1$, $\eta > 0$, $\beta \in [0, 1)$, and $\alpha \in [0, 1)$. The parameters are constant throughout the run, although they may depend on its prescribed horizon R. Initialize $\Delta_G^0 = 0$ and a deterministic global momentum m^0 (in particular, $m^0 = 0$ is allowed). At round r, all clients initialize $x_i^{r,0} = x^r$ and $m_i^{r,0} = m^r$, and apply

$$\begin{aligned} g_i^{r,k} &= \nabla f_i(x_i^{r,k}) + \xi_i^{r,k}, \\ m_i^{r,k+1} &= \beta m_i^{r,k} + (1-\beta) g_i^{r,k}, \\ q_i^{r,k} &= \mathrm{lmo}(m_i^{r,k+1}), \\ x_i^{r,k+1} &= x_i^{r,k} + \eta\big[(1-\alpha) q_i^{r,k} - \alpha \Delta_G^r\big]. \end{aligned} \tag{8}$$

The server sets

$$\begin{aligned} m^{r+1} &= \frac{1}{N} \sum_i m_i^{r,K}, \\ x^{r+1} &= \frac{1}{N} \sum_i x_i^{r,K}, \\ \Delta_G^{r+1} &= -\frac{x^{r+1} - x^r}{\eta K}. \end{aligned} \tag{9}$$

The momentum averaging in equation 9 is part of the algorithm and is essential to the tracking proof. This section treats exact LMO updates, uncompressed momentum, and no decoupled weight decay. Partial participation is treated separately in Section E.

Write

$$\delta = 1 - \beta, \quad H = \eta K, \quad \bar{m}^{r,k} = \frac{1}{N} \sum_i m_i^{r,k}, \quad \bar{q}^r = \frac{1}{NK} \sum_{i,k} q_i^{r,k}, \quad E_0 = \|m^0 - \nabla f(x^0)\|_\star.$$

Also define

$$\begin{aligned} B_K(\beta) &= \frac{1}{K} \sum_{j=1}^{K} (1 - \beta^j) \leq \min\{1, \delta(K+1)/2\}, \\ C_K &= 2\rho\sigma + 2L\eta(K-1) + \rho\zeta. \end{aligned} \tag{10}$$

C.2 DRIFT AND MOMENTUM CONSENSUS

Lemma C.1 (Bounded directions and virtual iterates)**.** *For every round and local step,*

$$\|\Delta_G^r\| \leq 1, \quad \|x_i^{r,k} - x^r\| \leq \eta k, \quad \|x^{r+1} - x^r\| \leq H.$$

Define the virtual iterate

$$y^r = x^r - \frac{\alpha H}{1-\alpha} \Delta_G^r.$$

Then $y^0 = x^0$ *and*

$$y^{r+1} = y^r + H\bar{q}^r, \qquad \|y^r - x^r\| \leq \frac{\alpha H}{1-\alpha}.$$

Proof. Averaging the local updates gives

$$\Delta_G^{r+1} = \alpha \Delta_G^r - (1-\alpha)\bar{q}^r.$$

Since $\|\bar{q}^r\| \leq 1$, induction from $\Delta_G^0 = 0$ gives the first bound. Every local mixed direction also has norm at most one, proving the local drift and global displacement bounds. Direct substitution yields

$$\begin{aligned} y^{r+1} - y^r &= -H\Delta_G^{r+1} - \frac{\alpha H}{1-\alpha}(\Delta_G^{r+1} - \Delta_G^r) \\ &= -\frac{H}{1-\alpha}(\Delta_G^{r+1} - \alpha \Delta_G^r) = H\bar{q}^r. \end{aligned}$$

The remaining claims follow from the definition. □

Lemma C.2 (Within-round momentum disagreement)**.** *For $j = 1, \dots, K$,*

$$\frac{1}{N}\sum_i \mathbb{E}\|m_i^{r,j} - \bar{m}^{r,j}\|_\star \le (1-\beta^j)C_K.$$

Consequently its average over j is at most $B_K(\beta)C_K$.

Proof. Let $\bar{g}^{r,k} = N^{-1}\sum_i g_i^{r,k}$ and $a^{r,k} = N^{-1}\sum_i \nabla f_i(x_i^{r,k})$. The conditional noise assumptions and Jensen's inequality imply

$$\frac{1}{N}\sum_i \mathbb{E}\|g_i^{r,k} - \bar{g}^{r,k}\|_\star \le 2\rho\sigma + \frac{1}{N}\sum_i \mathbb{E}\|\nabla f_i(x_i^{r,k}) - a^{r,k}\|_\star.$$

Insert $\nabla f_i(x^r)$ and $\nabla f(x^r)$ into the last difference. Smoothness and Lemma C.1 give

$$\frac{1}{N}\sum_i \mathbb{E}\|\nabla f_i(x_i^{r,k}) - a^{r,k}\|_\star \le 2L\eta k + \rho\zeta.$$

This bounds the expected gradient disagreement by C_K for all queried steps $k \le K-1$. The shared initialization implies the exact identity

$$m_i^{r,j} - \bar{m}^{r,j} = \delta\sum_{k=0}^{j-1}\beta^{j-1-k}(g_i^{r,k} - \bar{g}^{r,k}).$$

Taking norms, expectations, and averages proves the result, since $\delta\sum_{k=0}^{j-1}\beta^{j-1-k} = 1-\beta^j$. Finally, $1-\beta^j \le \min\{1, j\delta\}$ proves the bound for B_K. □

C.3 Global momentum tracking with synchronization included

Lemma C.3 (Post-update tracking of the round-start global gradient)**.** *The averaged post-update momentum satisfies*

$$\frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=0}^{K-1}\mathbb{E}\|\bar{m}^{r,k+1} - \nabla f(x^r)\|_\star \le \frac{E_0}{\delta KR} + L\eta(K-1) + \frac{L\eta}{\delta} + \rho\sigma\sqrt{\frac{\delta}{N(1+\beta)}}. \quad (11)$$

Proof. Set $T = KR$ and flatten $t = rK + k$. Define

$$z_t = \nabla f(x^{\lfloor t/K\rfloor}), \quad b_t = \frac{1}{N}\sum_i \nabla f_i(x_i^{r,k}) - z_t, \quad \nu_t = \frac{1}{N}\sum_i \xi_i^{r,k}.$$

Let $M_t = \bar{m}^{r,k}$ for $t = rK + k$. At a round boundary the averaged momentum is preserved exactly, because $\bar{m}^{r,K} = m^{r+1} = \bar{m}^{r+1,0}$. Therefore, with $M_T = m^R$, the single recursion

$$M_{t+1} = \beta M_t + \delta(z_t + b_t + \nu_t) \quad (0 \le t < T)$$

is valid even at synchronization boundaries. Moreover,

$$\|b_t\|_\star \le L\eta(K-1), \quad \mathbb{E}[\nu_t \mid \mathcal{F}_t] = 0, \quad \mathbb{E}[\|\nu_t\|_F^2 \mid \mathcal{F}_t] \le \sigma^2/N.$$

Here $\mathcal{F}_t$ is the same pre-query filtration with flattened indices. Unrolling the recursion and subtracting z_t gives the pathwise identity

$$M_{t+1} - z_t = \beta^{t+1}(M_0 - z_0) + \delta\sum_{s=0}^{t}\beta^{t-s}b_s + \delta\sum_{s=0}^{t}\beta^{t-s}\nu_s - \sum_{s=1}^{t}\beta^{t-s+1}(z_s - z_{s-1}). \quad (12)$$

The first term averaged over t is at most $E_0/(\delta T)$. The second term is at most $L\eta(K-1)$ for every t. For the third, martingale orthogonality gives

$$\mathbb{E}\Big\|\delta\sum_{s=0}^{t}\beta^{t-s}\nu_s\Big\|_\star \le \rho\left(\frac{\sigma^2\delta^2}{N}\sum_{s=0}^{t}\beta^{2(t-s)}\right)^{1/2} \le \rho\sigma\sqrt{\frac{\delta}{N(1+\beta)}}.$$

The martingale calculation uses deterministic geometric coefficients; it does not assume independence of gradients and iterates.

It remains to bound the last term, which explicitly represents round boundaries. The difference $z_s - z_{s-1}$ is zero unless $s = rK$, where $1 \le r \le R-1$. At each such boundary,

$$\|z_{rK} - z_{rK-1}\|_\star = \|\nabla f(x^r) - \nabla f(x^{r-1})\|_\star \le LH.$$

Hence

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\Big\|\sum_{s=1}^{t}\beta^{t-s+1}(z_s - z_{s-1})\Big\|_\star \le \frac{LH}{T}\sum_{r=1}^{R-1}\sum_{t=rK}^{T-1}\beta^{t-rK+1} \le \frac{LH(R-1)}{\delta T} \le \frac{L\eta}{\delta}.$$

Combining these four bounds proves the claim. In particular, no claim that each individual client iterate moves by at most η across a server synchronization is used. □

C.4 Main theorem and finite-horizon rates

Theorem C.4 (`FMuon` finite-horizon stationarity bound)**.** *Under Assumptions A.1–A.3, the algorithm equation 8–equation 9 with $\Delta_G^0 = 0$ satisfies, for every $0 \le \alpha < 1$,*

$$\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}G(x^r) \le \frac{\Delta}{\eta KR} + L\eta K\left(\frac{1}{2} + \frac{\alpha}{1-\alpha}\right) + \frac{2E_0}{\delta KR} + 2L\eta(K-1) + \frac{2L\eta}{\delta} + 2\rho\sigma\sqrt{\frac{\delta}{N(1+\beta)}} + 2B_K(\beta)C_K. \tag{13}$$

All constants are explicit, and $\delta = 1 - \beta > 0$.

Proof. Let

$$D_r = \frac{1}{NK}\sum_{i,k}\mathbb{E}\|m_i^{r,k+1} - \nabla f(x^r)\|_\star.$$

Lemma A.4 gives

$$\mathbb{E}\langle\nabla f(x^r), \bar{q}^r\rangle \le -\mathbb{E}G(x^r) + 2D_r.$$

Use the virtual sequence in Lemma C.1. By smoothness and $\|\bar{q}^r\| \le 1$,

$$f(y^{r+1}) \le f(y^r) + H\langle\nabla f(y^r), \bar{q}^r\rangle + \frac{LH^2}{2} \le f(y^r) + H\langle\nabla f(x^r), \bar{q}^r\rangle + LH^2\frac{\alpha}{1-\alpha} + \frac{LH^2}{2}.$$

Take expectations and sum. Since $y^0 = x^0$ and $f(y^R) \ge f^\star$,

$$\frac{1}{R}\sum_{r}\mathbb{E}G(x^r) \le \frac{\Delta}{HR} + LH\left(\frac{1}{2} + \frac{\alpha}{1-\alpha}\right) + \frac{2}{R}\sum_{r}D_r.$$

The triangle inequality splits each term of D_r into

$$\|m_i^{r,k+1} - \bar{m}^{r,k+1}\|_\star + \|\bar{m}^{r,k+1} - \nabla f(x^r)\|_\star.$$

Apply Lemmas C.2 and C.3 to complete the proof. □

Corollary C.5 (Vanishing stochastic stationarity error, including $\alpha = 1/2$)**.** *Fix K, N, $\alpha \in [0, 1)$, and $\eta_0 > 0$, independently of R. For each prescribed horizon R, use throughout that run*

$$\eta = \frac{\eta_0}{KR^{3/4}}, \qquad \delta = 1 - \beta = R^{-1/2}.$$

Then

$$\begin{aligned}\frac{1}{R}\sum_r \mathbb{E}G(x^r) \le &\left(\frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \frac{2\rho\sigma}{\sqrt{N}}\right) R^{-1/4} \\ &+ \left(\frac{2E_0}{K} + \rho(K+1)(2\sigma+\zeta)\right) R^{-1/2} \\ &+ L\eta_0\left(\frac{1}{2} + \frac{\alpha}{1-\alpha} + \frac{2(K-1)}{K}\right) R^{-3/4} \\ &+ \frac{2L\eta_0(K+1)(K-1)}{K} R^{-5/4}. \end{aligned} \tag{14}$$

In particular, the expected dual-gradient norm of an independently, uniformly sampled round iterate tends to zero at rate $O(R^{-1/4})$. There is no non-vanishing noise or heterogeneity residual in this bound.

Proof. Use $2B_K \le \delta(K+1)$ and $(1+\beta)^{-1/2} \le 1$ in Theorem C.4, then substitute the parameters. If U is uniform on $\{0, \dots, R-1\}$ and independent of the run, $\mathbb{E}G(x^U) = R^{-1}\sum_r \mathbb{E}G(x^r)$. □

Corollary C.6 (Deterministic rate)**.** *If $\sigma = 0$, fix $K, N, \alpha < 1$, and use $\eta = \eta_0/(KR^{2/3})$ and $\delta = R^{-1/3}$. Then*

$$\begin{aligned}\frac{1}{R}\sum_r G(x^r) \le &\left(\frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \rho(K+1)\zeta\right) R^{-1/3} \\ &+ \left[\frac{2E_0}{K} + L\eta_0\left(\frac{1}{2} + \frac{\alpha}{1-\alpha} + \frac{2(K-1)}{K}\right)\right] R^{-2/3} \\ &+ \frac{2L\eta_0(K+1)(K-1)}{K} R^{-1}. \end{aligned}$$

Thus the deterministic average stationarity measure is $O(R^{-1/3})$.

Proof. Set $\sigma = 0$ in the main theorem and substitute the stated parameters. □

Remark C.7 (The original schedule). For the original schedule $\eta = \eta_0/(K\sqrt{R})$ and $1 - \beta = R^{-1/4}$, the refined stochastic term is proportional to $\sigma R^{-1/8}/\sqrt{N}$. For fixed $K, N, \alpha < 1$, our bound is then $O(R^{-1/8})$ when $\sigma > 0$, and $O(R^{-1/4})$ when $\sigma = 0$. The $O(R^{-1/4})$ stochastic result above therefore uses a different, explicitly stated schedule. Neither corollary justifies exact convergence under a fixed practical momentum coefficient such as $\beta = 0.95$ or 0.98. These are finite-horizon guarantees for a family of runs, not a proof for substituting a time-varying β_r into the same formulas.

C.5 Full participation without a uniform heterogeneity assumption

The shared initialization also yields a weaker-in-constants guarantee without a global bound on client dissimilarity. This result is useful when uniform heterogeneity is unrealistic; it does not import the FedAvg-M proof or identify the two algorithms.

Theorem C.8 (Convergence without uniformly bounded heterogeneity)**.** *Suppose only Assumptions A.1 and A.2 hold, together with $f^\star > -\infty$. Define the finite initial disagreement*

$$J_0 = \frac{1}{N}\sum_i \|\nabla f_i(x^0) - \nabla f(x^0)\|_\star.$$

Under full participation, the main bound equation 13 remains valid after replacing C_K by

$$C_{K,R} = 2\rho\sigma + 2L\eta(K-1) + J_0 + 2L\eta K(R-1).$$

For fixed $K, N, \alpha < 1$ and $\eta_0 > 0$, choose $\eta = \eta_0/(KR^{3/4})$ and $\delta = R^{-1/2}$. Then

$$\frac{1}{R}\sum_r \mathbb{E}G(x^r) \le A_0 R^{-1/4} + A_1 R^{-1/2} + A_2 R^{-3/4} + A_3 R^{-5/4}, \tag{15}$$

$$A_0 = \frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \frac{2\rho\sigma}{\sqrt{N}} + 2L\eta_0(K+1),$$
$$A_1 = \frac{2E_0}{K} + (K+1)(2\rho\sigma + J_0),$$
$$A_2 = L\eta_0\left(\frac{1}{2} + \frac{\alpha}{1-\alpha} + \frac{2(K-1)}{K}\right),$$
$$A_3 = \frac{2L\eta_0(K+1)(K-1)}{K}.$$

In particular, the expected average stationarity measure is $O(R^{-1/4})$ without Assumption A.3.

Proof. The bounded-step lemma holds without any heterogeneity assumption, so $\|x^r - x^0\| \le rH$. By smoothness of each f_i and of f,

$$\frac{1}{N}\sum_i \|\nabla f_i(x^r) - \nabla f(x^r)\|_\star \le J_0 + \frac{1}{N}\sum_i \|\nabla f_i(x^r) - \nabla f_i(x^0)\|_\star + \|\nabla f(x^r) - \nabla f(x^0)\|_\star$$
$$\le J_0 + 2LrH \le J_0 + 2LH(R-1).$$

Use this bound in the momentum-consensus lemma in place of $\rho\zeta$. The tracking lemma and the virtual-iterate proof do not use bounded heterogeneity, so the same main inequality follows with $C_{K,R}$. Substitute the stated parameters and use $2B_K \le \delta(K+1)$. The additional growing-disagreement term is bounded by

$$2B_K\, 2LH(R-1) \le 2L\eta_0(K+1)(R-1)R^{-5/4} \le 2L\eta_0(K+1)R^{-1/4}.$$

The remaining terms are exactly those in the earlier substitution with $\rho\zeta$ replaced by J_0. This proves the explicit rate. □

Remark C.9 (Scope of assumption removal). J_0 is evaluated only at the fixed initial point; no uniform bound over all parameters is imposed. The result allows disagreement to grow with the length of the trajectory and compensates through the horizon-dependent momentum coefficient. Its constants grow with K; it is not a linear speedup statement in N or K. The sharper deterministic $R^{-1/3}$ corollary above still invokes uniform bounded heterogeneity. The same assumption removal is not claimed for the partial-participation result.

D A STRICT, SAME-BUDGET SEPARATION FROM LOCAL MUON

For an algorithm A, write

$$\mathcal{G}_R(A) = \frac{1}{R}\sum_{r=0}^{R-1} \mathbb{E}\|\nabla f(x_A^r)\|_\star.$$

All comparisons below use this same stationarity measure, the same objective, the same initial model, the same number of participating clients and local steps, and the same local gradient-query budget.

Theorem D.1 (Strict finite-horizon separation on a heterogeneous matrix family)**.** *Consider the family in Proposition B.4. Fix any $d, K \ge 1$, $c > 0$, $\alpha \in [0, 1)$, and $0 < \eta_0 < c$. For each horizon R, run both algorithms with*

$$\eta_R = \frac{\eta_0}{KR^{2/3}}, \qquad \beta_R = 1 - R^{-1/3}.$$

Initialize `FMuon` *with $m^0 = 0$ and $\Delta_G^0 = 0$. For either Local Muon variant in the proposition,*

$$\mathcal{G}_R(\text{Local Muon}) = dc, \qquad \mathcal{G}_R(\texttt{FMuon}) \le C_* R^{-1/3},$$

where the explicit finite constant is

$$C_* = \frac{\Delta}{\eta_0} + \frac{2L\eta_0}{K} + \rho(K+1)\zeta + \frac{2E_0}{K} + L\eta_0\left(\frac{1}{2} + \frac{\alpha}{1-\alpha} + \frac{2(K-1)}{K}\right) + \frac{2L\eta_0(K+1)(K-1)}{K}, \tag{16}$$

with $L = d$, $\rho = \sqrt{d}$, $\zeta = 2c\sqrt{d}$, $\Delta = dc^2/2$, *and* $E_0 = dc$. *In particular, for*

$$R \geq \max\left\{1, \left\lceil \left(\frac{2C_*}{dc}\right)^3 \right\rceil\right\},$$

`FMuon` *has strictly smaller average stationarity measure:*

$$\mathcal{G}_R(\texttt{FMuon}) \leq dc/2 < \mathcal{G}_R(\text{Local Muon}).$$

Both runs use exactly $2KR$ *local gradient queries.*

Proof. The condition $K\eta_R = \eta_0 R^{-2/3} < c$ holds at every horizon. The Local Muon proposition therefore gives its exact error dc, independently of the chosen β_R. For FMuon, apply Corollary C.6 to the same objectives and the same η_R, β_R. For $R \geq 1$, both $R^{-2/3}$ and R^{-1} are at most $R^{-1/3}$, giving the constant C_*. The displayed threshold then implies the strict inequality. The reset Local Muon implementation uses one initial gradient and $K-1$ subsequent gradients per client and round; the retained implementation and FMuon use K gradients. Thus all have exactly the same query budget. □

Remark D.2 (The logical content of superiority). This is a constructive separation between an exact positive lower bound for Local Muon and a vanishing upper bound for FMuon on the same problem family. It does not compare two unrelated upper bounds, and it does not rely on momentum resetting as the sole reason for baseline failure. The statement is about rounds and gradient queries, not equal transmitted bytes or wall-clock time. A uniform dominance statement over all objectives is false, as the next proposition demonstrates.

Proposition D.3 (No universal finite-horizon dominance)**.** *Fix* $\alpha \in (0,1)$ *and* $\beta \in [0,1)$. *There is a homogeneous, deterministic example on which the two-round average stationarity measure of Local Muon is strictly smaller than that of FMuon, using the same stepsize and gradient-query budget.*

Proof. Let all clients have $f_i(x) = x^2/2$, choose $K = 1$, $x^0 = a > 0$, and $\eta = a$. Use zero initial global momentum and direction for FMuon. The first queried gradient is positive, hence the first exact LMO is -1 for both algorithms. Local Muon gives $x^1_{\text{Local}} = 0$, whereas FMuon gives $x^1_{\text{Fed}} = \alpha a$. Consequently, for the two-round metric using $r = 0, 1$,

$$\mathcal{G}_2(\text{Local Muon}) = a/2 < (1+\alpha)a/2 = \mathcal{G}_2(\text{FMuon}).$$

The argument also applies to retained Local Muon initialized from zero with the gradient update preceding the LMO. The second round's update does not change this metric, which uses the two round-start iterates. □

Proposition D.4 (A fixed momentum coefficient can leave a genuine residual)**.** *For every fixed* $\beta \in [0,1)$, $\alpha \in [0,1)$, *and* $\eta > 0$, *there is a deterministic two-client scalar problem satisfying all three assumptions on which FMuon with* $K = 1$, $m^0 = 0$, *and* $x^0 = \Delta^0_G = 0$ *satisfies*

$$G(x^r) = \frac{(1-\beta)\zeta}{2} > 0 \quad \textit{for all } r.$$

Proof. Fix $z > 0$, set $\delta = 1 - \beta$ and $c = \delta z/2$, and let

$$f_1(x) = \tfrac{1}{2}(x + c + z)^2, \qquad f_2(x) = \tfrac{1}{2}(x + c - z)^2.$$

Then $\nabla f(x) = x + c$, $L = \rho = 1$, and $\zeta = z$. Suppose $x^r = \Delta^r_G = 0$ and $m^r = (1-\beta^r)c$. The two post-gradient momenta are

$$m^{r,1}_1 = (1-\beta^{r+1})c + \delta z > 0, \qquad m^{r,1}_2 = (1-\beta^{r+1})c - \delta z < 0.$$

The second inequality follows from $c < \delta z$. Their LMO directions cancel, so both the model and global direction remain zero, while their average momentum is $(1-\beta^{r+1})c$. Induction proves the claim. Thus fixed-β exact convergence cannot be asserted for this algorithm under the stated assumptions. The objective in this example depends on the fixed coefficient; it does not refute the horizon-scheduled results on a fixed objective. □

E UNIFORM PARTIAL PARTICIPATION

We now analyze our algorithm with sampled clients. Assume $N \geq 2$ and sample a uniform subset $S_r \subset [N]$ of fixed size $S \in \{1, \dots, N\}$ without replacement at each round, conditionally on the entire past. Only sampled clients compute updates; each starts from the current shared x^r, m^r, Δ_G^r, regardless of its previous participation. The server averages their endpoints and momenta using $1/S$:

$$x^{r+1} = \frac{1}{S} \sum_{i \in S_r} x_i^{r,K}, \qquad m^{r+1} = \frac{1}{S} \sum_{i \in S_r} m_i^{r,K}, \qquad \Delta_G^{r+1} = -\frac{x^{r+1} - x^r}{\eta K}.$$

The local rule is unchanged. The conditional oracle assumption holds for selected clients, with a pre-query filtration containing S_r.

Define the finite-population factors

$$p_S = \frac{N - S}{S(N-1)}, \qquad \chi_S = 1 - p_S = \frac{N(S-1)}{S(N-1)}, \qquad a_K = 1 - \beta^K \leq \min\{1, K\delta\},$$

and $C_{K,S} = 2\rho\sigma + 2L\eta(K-1) + \rho\sqrt{\chi_S}\zeta$.

Theorem E.1 (Stationarity under uniform client sampling)**.** *Under all three assumptions, the sampled algorithm satisfies*

$$\frac{1}{R} \sum_r \mathbb{E} G(x^r) \leq \frac{\Delta}{\eta K R} + L\eta K \left(\frac{1}{2} + \frac{\alpha}{1-\alpha} \right) + \frac{2E_0}{\delta K R} + 2L\eta(K-1) + \frac{2L\eta}{\delta} + 2\rho\sigma \sqrt{\frac{\delta}{S(1+\beta)}} + 2\rho\zeta\sqrt{p_S a_K} + 2B_K(\beta) C_{K,S}. \tag{17}$$

For fixed $N, S, K, \alpha < 1$, the schedule $\eta = \eta_0/(KR^{3/4})$, $\delta = R^{-1/2}$ gives $\mathcal{G}_R = O(R^{-1/4})$. Its client-sampling term is at most $2\rho\zeta\sqrt{K p_S}\, R^{-1/4}$ and vanishes identically when $S = N$.

Proof. The unit-direction and virtual-iterate identities remain true when the averages are over S_r. We prove the two modified tracking bounds.

Step 1: finite-population identities. Condition on the history $\mathcal{H}_r$ immediately before sampling, so that x^r and the local gradients at x^r are fixed. Write $v_i = \nabla f_i(x^r) - \nabla f(x^r)$ and $v_{S_r} = S^{-1} \sum_{i \in S_r} v_i$. The identities $\sum_i v_i = 0$, $\mathbb{P}(i \in S_r) = S/N$, and $\mathbb{P}(i, j \in S_r) = S(S-1)/(N(N-1))$ give

$$\mathbb{E}[\|v_{S_r}\|_F^2 \mid \mathcal{H}_r] = p_S \frac{1}{N} \sum_i \|v_i\|_F^2 \leq p_S \zeta^2,$$

$$\mathbb{E}\left[\frac{1}{S} \sum_{i \in S_r} \|v_i - v_{S_r}\|_F^2 \,\middle|\, \mathcal{H}_r \right] = \chi_S \frac{1}{N} \sum_i \|v_i\|_F^2 \leq \chi_S \zeta^2.$$

The first follows by expanding the square; the second subtracts the sample-mean square from the sample mean of squares.

Step 2: consensus among selected clients. Add and subtract the sampled gradients at x^r. Smoothness bounds their subsequent displacement by $2L\eta k$, whereas the second identity above and Jensen's inequality bound their initial centered disagreement by $\rho\sqrt{\chi_S}\zeta$. The noise contribution is at most $2\rho\sigma$. Unrolling the shared-initialization recursion therefore gives

$$\frac{1}{K} \sum_{j=1}^{K} \mathbb{E}\left[\frac{1}{S} \sum_{i \in S_r} \|m_i^{r,j} - \bar{m}^{r,j}\|_\star \right] \leq B_K C_{K,S}.$$

Here $\bar{m}^{r,j}$ denotes the sampled-client average.

Step 3: sampled global tracking. Flatten $t = rK + k$ as before and let $z_t = \nabla f(x^r)$. Define

$$b_t = \frac{1}{S} \sum_{i \in S_r} (\nabla f_i(x_i^{r,k}) - \nabla f_i(x^r)), \qquad \nu_t = \frac{1}{S} \sum_{i \in S_r} \xi_i^{r,k}.$$

The averaged momentum obeys

$$M_{t+1} = \beta M_t + \delta(z_t + b_t + v_{S_r} + \nu_t).$$

We still have $\|b_t\|_\star \le L\eta(K-1)$ and $\mathbb{E}[\|\nu_t\|_F^2 \mid \mathcal{F}_t] \le \sigma^2/S$. Thus initialization, local drift, gradient-query noise, and round-boundary target changes contribute exactly the terms in the full-participation tracking bound with N replaced by S. The new sampling contribution is

$$Z_t = \delta \sum_{s=0}^{t} \beta^{t-s} v_{S_{\lfloor s/K \rfloor}} = \sum_{\ell=0}^{\lfloor t/K \rfloor} w_{t,\ell} v_{S_\ell},$$

where

$$w_{t,\ell} = \delta \sum_{s=\ell K}^{\min\{t,(\ell+1)K-1\}} \beta^{t-s}.$$

The coefficients are deterministic, nonnegative, their sum is at most one, and each is at most a_K. Because $\mathbb{E}[v_{S_\ell} \mid \mathcal{H}_\ell] = 0$, sampling errors from distinct rounds are orthogonal in expectation, although each is reused within its round. Consequently,

$$\mathbb{E}\|Z_t\|_F^2 \le p_S \zeta^2 \sum_\ell w_{t,\ell}^2 \le p_S \zeta^2 a_K, \qquad \mathbb{E}\|Z_t\|_\star \le \rho\zeta\sqrt{p_S a_K}.$$

Combining the separate terms by the triangle inequality is valid without assuming independence between sampling errors and gradient-query noise. It follows that the averaged tracking error is at most

$$\frac{E_0}{\delta K R} + L\eta(K-1) + \frac{L\eta}{\delta} + \rho\sigma\sqrt{\frac{\delta}{S(1+\beta)}} + \rho\zeta\sqrt{p_S a_K}.$$

Combining the virtual-iterate argument with Step 2 establishes the stated bound equation 17. Finally use $a_K \le K\delta$, $2B_K \le (K+1)\delta$, and substitute the schedule. Every term tends to zero for fixed N, S, K, α. □

Remark E.2 (Why replacing N by S is insufficient). The term $2\rho\zeta\sqrt{p_S a_K}$ is distinct from gradient-query noise. Treating the K queries within a sampled round as K independent client samples would incorrectly remove this contribution. This result also assumes every selected client receives the current shared state; stale client states and their communication cost are separate issues. The strict Local Muon separation above is a full-participation result; it is not automatically a lower bound for sampled Local Muon.

F Relation to the supplied references and limits of the claims

Different algorithms must have different proofs. Takezawa et al. (ICLR 2026, Algorithm 1, p. 4) apply the LMO to a control-variate-corrected momentum $m_i - c_i + c$ and retain local states. Our algorithm applies the LMO to the local momentum, shares the round-end momentum, and mixes the LMO output with a historical direction. These operations do not commute. Their Theorem 2 cannot be invoked as a convergence theorem for our update. Their lower-bound construction motivates the type of comparison, but our same-instance separation must use the algorithm-specific upper bound proved here.

Cheng et al., *Momentum Benefits Non-IID Federated Learning Simply and Provably* (ICLR 2024, Algorithm 1 and Theorem 1, p. 5), mix a raw local gradient with a shared estimate using a coefficient denoted β in that paper and use separate local and global stepsizes. That coefficient is not our retained-momentum coefficient. Their linear gradient-mixture analysis does not carry over by replacing the gradient with a nonlinear LMO. In particular, their squared Euclidean-gradient criterion cannot be directly compared numerically to our dual-gradient norm. Our Theorem C.8 removes uniform heterogeneity through an explicit trajectory bound and a specified horizon schedule; it does not claim their constants or linear speedup.

Which module is justified by which argument? Shared initialization eliminates the inherited term in the identity

$$m_i^{r,j} - \bar{m}^{r,j} = \beta^j (m_i^{r,0} - \bar{m}^{r,0}) + \delta \sum_{k=0}^{j-1} \beta^{j-1-k} (g_i^{r,k} - \bar{g}^{r,k}).$$

This produces the vanishing factor B_K. The virtual iterate proves that history-direction mixing is compatible with convergence for every fixed $\alpha < 1$, including $\alpha = 1/2$. It does not establish that $\alpha > 0$ is theoretically better than $\alpha = 0$; the bounds also hold at zero. Comparing individual heterogeneity coefficients in two upper bounds is not a proof that the actual errors are ordered.

Uncovered implementations and metrics. The constants depend on K and $(1-\alpha)^{-1}$. Rates stated for fixed K, α must not hide their growth with R. The guarantees concern expected dual-gradient norm, not expected squared Frobenius-gradient norm, last-iterate convergence, or convergence of the entire parameter sequence. SVD compression changes the momentum recursion and requires an explicit compression-error analysis. Decoupled weight decay and time-varying within-run learning rates are also outside the stated algorithm. None of the theoretical comparisons proves lower total communication or wall-clock time.

Conditional inexact feasible LMO extension. If $\|q_i^{r,k}\| \le 1$ and

$$\langle m_i^{r,k+1}, q_i^{r,k} \rangle \le -\|m_i^{r,k+1}\|_\star + \epsilon_i^{r,k}, \qquad \epsilon_i^{r,k} \ge 0,$$

the perturbation inequality acquires the additive error $\epsilon_i^{r,k}$. Every corresponding main bound gains

$$\frac{1}{RNK} \sum_{r,i,k} \mathbb{E}\epsilon_i^{r,k} \quad \text{under full participation,}$$

or $(RSK)^{-1} \sum_{r,k} \mathbb{E} \sum_{i \in S_r} \epsilon_i^{r,k}$ under sampling. This follows by inserting the inexact inequality into the same descent argument. No assertion is made that an unmodified NS-5 implementation is feasible or has vanishing additive oracle error.

Suggested main-text claim. We establish an algorithm-specific convergence guarantee for shared-momentum FMuon with every fixed history-mixing coefficient in $[0, 1)$. Under a specified horizon-dependent schedule, the expected average dual-gradient norm vanishes at rate $O(R^{-1/4})$ under stochastic gradients. Full participation admits this result without uniformly bounded heterogeneity; uniform partial participation admits it under bounded heterogeneity with an explicit client-sampling term. On a heterogeneous matrix quadratic family, reset and retained Local Muon both remain at a nonstationary point, while FMuon attains arbitrarily small average stationarity error with the same local gradient-query budget. This establishes a strict separation on that family, rather than universal finite-horizon dominance.

Table 6: A detailed summary of CIFAR-100 and Tiny ImageNet: number of classes, image size, and dataset splits.

Dataset	#Classes	Image Size	Train	Val	Test	Total	Train / class
CIFAR-100	100	$3 \times 32 \times 32$	50,000	—	10,000	60,000	500
Tiny ImageNet	200	$3 \times 64 \times 64$	100,000	10,000	10,000	120,000	500

Notes. (1) CIFAR-100 provides no official validation split; a subset of the training set is commonly reserved as dev/val.
(2) CIFAR-100 contains 100 fine-grained classes; 20 coarse superclasses are also defined for hierarchical labeling.
(3) Tiny ImageNet is a subset of ImageNet synsets: per class 500 train, 50 val, and 50 test images (test labels are not publicly released).
(4) Both datasets are single-label classification with RGB images resized to fixed resolutions.

G Experimental Setup

G.1 Vision Datasets

We evaluate our methods on two widely-used benchmark datasets in federated learning: **CIFAR-100** and **Tiny ImageNet**.

- **CIFAR-100** (Krizhevsky et al., 2009): Contains 100 classes with 600 color images per class at a resolution of 32×32. It is a standard benchmark for evaluating federated image classification methods.
- **Tiny ImageNet**: A subset of ImageNet with 200 classes and 500 training images per class, providing a more challenging and high-resolution classification task.

G.2 Federated Learning Configuration

We simulate a cross-device federated learning environment using the following settings:

Table 7: Hyperparameter configuration of ResNet-18 and ViT-Tiny (CIFAR-100) across different algorithms.

Method	Local Optimizer	Local LR	α	β_1	β_2	Weight Decay
FedAvg (Local SGD)	SGD	0.1	—	—	—	0.001
FedProx	SGD	0.1	—	—	—	0.001
FedDyn	SGD	0.1	—	—	—	0.001
Mime	SGD	0.1	—	—	—	0.001
FedAdam	SGD	0.1	—	0.9	0.98	0.001
SCAFFOLD	SGD	0.1	—	—	—	0.001
FedCM	SGD	0.1	0.9	—	—	0.001
FedLADA	AdamW	3e-4	0.9	0.9	0.999	0.01
Local AdamW	AdamW	3e-4	—	0.9	0.999	0.01
FedMuon	Muon	3e-2	—	0.95	—	0.01
Local Muon	Muon	3e-2	—	0.95	—	0.01
FMuon	Muon	3e-2	0.5	0.95	—	0.01

Note: $\alpha = 0.5$ is the alignment coefficient for FMuon only. Both FMuon and Local Muon use momentum coefficient $\beta = 0.95$. Learning rates and weight decay are method-specific, as listed above.

- **Number of clients:** 100
- **Client participation rate:** 10% per round
- **Communication rounds:** 300
- **Local update steps:** 50 iterations
- **Batch size:** 50

Table 8: Hyperparameter configuration of RoBERTa-base fine-tuning across different algorithms.

Method	Local Optimizer	Local LR	α	β_1	β_2	Weight Decay
FedAvg (Local SGD)	SGD	0.1	—	—	—	0.001
FedProx	SGD	0.1	0.01	—	—	0.001
FedDyn	SGD	0.1	0.01	—	—	0.001
Mime	SGD	0.1	—	—	—	0.001
FedAdam	SGD	0.1	—	0.9	0.98	0.001
SCAFFOLD	SGD	0.1	—	—	—	0.001
FedCM	SGD	0.1	0.9	—	—	0.001
FedLADA	AdamW	1e-4	0.9	0.9	0.999	0.01
Local AdamW	AdamW	1e-4	—	0.9	0.999	0.01
FedMuon	Muon	1e-3	—	0.95	—	0.01
Local Muon	Muon	1e-3	—	0.95	—	0.01
FMuon	Muon	1e-3	0.5	0.95	—	0.01

Note: $\alpha = 0.5$ is the alignment coefficient for FMuon only. Both FMuon and Local Muon use momentum coefficient $\beta = 0.95$. Learning rates and weight decay are method-specific, as listed above.

Table 9: Hyperparameter configuration of ViT-Base and Swin Transformer-Tiny fine-tuning across different algorithms.

Method	Local Optimizer	Local LR	α	β_1	β_2	Weight Decay
FedAvg (Local SGD)	SGD	0.1	—	—	—	0.001
FedProx	SGD	0.1	0.01	—	—	0.001
FedDyn	SGD	0.1	0.01	—	—	0.001
Mime	SGD	0.1	—	—	—	0.001
FedAdam	SGD	0.1	—	0.9	0.98	0.001
SCAFFOLD	SGD	0.1	—	—	—	0.001
FedCM	SGD	0.1	0.9	—	—	0.001
FedLADA	AdamW	1e-4	0.9	0.9	0.999	0.01
Local AdamW	AdamW	1e-4	—	0.9	0.999	0.01
FedMuon	Muon	1e-3	—	0.95	—	0.01
Local Muon	Muon	1e-3	—	0.95	—	0.01
FMuon	Muon	1e-3	0.5	0.95	—	0.01

Note: $\alpha = 0.5$ is the alignment coefficient for FMuon only. Both FMuon and Local Muon use momentum coefficient $\beta = 0.95$. Learning rates and weight decay are method-specific, as listed above.

We perform grid search to tune the learning rates for each algorithm:

- **FedAvg**, **SCAFFOLD**, **FedCM**, and **FedAdam** use a local learning rate of `0.1`.
- **FedLADA**, **Local AdamW**, use a local learning rate of `3e-4`.
- **FMuon** and **Local Muon** use a local learning rate of `3e-2`, $\beta = 0.95$ ($\alpha = 0.5$ for FMuon only)

G.3 ResNet-18 Architecture

We adopt **ResNet-18** as the backbone model. To better adapt it to CIFAR-100, we modify its architecture following standard practices (He et al., 2016):

- Replace the original 7×7 convolution with a 3×3 kernel.
- Remove the initial downsampling layers (stride-2 convolution and max pooling).

We also compare **Batch Normalization (BN)** and **Group Normalization (GN)** in ResNet-18. Empirically, BN outperforms GN on CIFAR-100, so we adopt the BN-based version, denoted as **ResNet-18-BN**, throughout our experiments.

G.4 SETTING FOR VIT-TINY

We construct a lightweight Vision Transformer model, **ViT-Tiny**, specifically tailored for federated learning on the CIFAR-100 dataset. The design is based on the standard ViT architecture (Dosovitskiy et al., 2020), with modifications to accommodate the small input size and limited data per client.

- **Input resolution:** 32×32
- **Patch size:** 4×4, resulting in 64 tokens per image
- **Embedding dimension:** 192
- **Number of Transformer layers:** 6
- **Number of attention heads:** 3
- **Normalization:** LayerNorm (applied before attention and MLP blocks)
- **Classification head:** Linear projection to 100 classes (CIFAR-100)
- **Activation:** GELU
- **Initialization:** Xavier/Glorot for linear layers; sinusoidal positional encoding

To regularize training, we apply dropout (0.1) to both attention and MLP layers. The ViT-Tiny models in this setting are trained from scratch.

Federated Learning Configuration. We adopt the same federated learning setup as used in our ResNet experiments for a fair comparison:

- **Number of clients:** 100
- **Client participation rate:** 10%
- **Communication rounds:** 300
- **Local update steps:** 50 iterations per round
- **Local batch size:** 50

Learning Rate Schedule. We perform grid search to identify optimal learning rates for each algorithm:

- **FedAvg**, **SCAFFOLD**, **FedCM**, and **FedAdam**: local learning rate of `0.1`
- **FedLADA**, **Local AdamW**: local learning rate of `3e-4`
- **FMuon** and **Local Muon**: local learning rate of `3e-2`, $\beta = 0.95$ ($\alpha = 0.5$ for FMuon only)

Weight Decay. To ensure fair comparison under different regularization settings, we assign:

- **FedAvg**, **SCAFFOLD**, **FedCM**, **FedAdam**: weight decay = `0.001`
- **Local Muon**, **FedLADA**, **Local AdamW**, **FMuon**: weight decay = `0.01`

G.5 VISION TRANSFORMER FINE-TUNING SETTINGS

To demonstrate the effectiveness of our method on large-scale vision models, we conduct fine-tuning experiments using **Swin Transformer-Tiny** and **ViT-Base** on **Tiny ImageNet** and **CIFAR-100**. For both models, we initialize from official `ImageNet-22K` pre-trained weights (Liu et al., 2021; Dosovitskiy et al., 2020) to ensure consistency across methods.

Data Preprocessing. To align with the input resolution required by Swin and ViT, we resize images from both datasets to 224×224 using bilinear interpolation. Standard data augmentation techniques such as random cropping, horizontal flipping, and RandAugment are applied locally at the client side.

Table 10: Summary of GLUE datasets: task type, number of classes, and training-set sizes.

Dataset	Task Type	#Classes	Train Size
MNLI	Natural Language Inference (entailment)	3	392,702
SST-2	Sentiment Classification (binary)	2	67,349
MRPC	Paraphrase Detection (binary)	2	3,668
CoLA	Linguistic Acceptability (binary)	2	8,551
QNLI	Question-Answer NLI (binary)	2	104,743
QQP	Duplicate Question Detection (binary)	2	363,846
RTE	Recognizing Textual Entailment (binary)	2	2,490

Table 11: Hyperparameter configuration for RoBERTa-Base with LoRA across GLUE tasks. LoRA is applied with $r = 16$ to both query and value projections. Method-specific optimizers are listed in Table 8.

Method	Setting	GLUE Tasks						
		MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE
RoBERTa-Base + LoRA	Batch size	16	16	16	16	16	16	16
	Max seq. length	128	128	128	128	128	128	128
	LoRA ranks ($r_q = r_v$)	16	16	16	16	16	16	16
	LoRA scaling α	32	32	32	32	32	32	32
	Dropout	0.1						

Federated Learning Configuration. To simulate a realistic cross-device setting, we configure:

- **Number of clients:** 100
- **Client participation rate:** 5% per communication round
- **Communication rounds:** 100
- **Local update steps:** 50 iterations
- **Batch size:** 16

Learning Rate Configuration. We apply grid search to find optimal learning rates and use **cosine learning rate decay** with no warmup unless otherwise stated:

- **FedAvg**, **SCAFFOLD**, **FedCM**: local LR = `0.1`
- **FedLADA**, **Local AdamW**: local LR = `1e-4`
- **FMuon** and **Local Muon**: local LR = `1e-3`

Weight Decay. To ensure fair comparison under different regularization settings, we assign:

- **FedAvg**, **SCAFFOLD**, **FedCM**: weight decay = `0.001`
- **Local Muon**, **FedLADA**, **Local AdamW**, **FMuon**: weight decay = `0.01`

Remarks. We find that Swin Transformer benefits from hierarchical attention and DropPath when training with limited local data. Our method shows stable convergence and avoids loss spikes often seen in large-scale federated fine-tuning. All models are implemented using the HuggingFace Transformers and Timm libraries.

G.6 RoBERTa-Base Fine-Tuning Settings

We fine-tune the **RoBERTa-Base** model using **LoRA** (Low-Rank Adaptation) on a subset of the GLUE benchmark. The LoRA adaptation is applied to the query and value projection matrices of the self-attention modules. The following table summarizes the hyperparameter settings used across tasks.

Explanation. We use a uniform batch size of 16 and sequence length of 128 across all tasks. LoRA is configured with a rank of 16 and scaling factor $\alpha = 32$. Dropout is set to 0.1. The LoRA factors are trained while the pretrained backbone remains frozen. Optimizers and weight decay follow Table 8.

G.7 Federated Language-Model Settings

To evaluate our algorithm under a smaller-scale federation, we further conduct experiments with a reduced number of clients and adjusted participation parameters.

Federated Setup. We simulate a federated learning environment with the following configuration:

- **Number of clients:** 4 or 20
- **Client participation rate:** 100% for 4 clients or 20% for 20 clients (4 participating clients per round)
- **Communication rounds:** 100
- **Local update steps:** 50
- **Local batch size:** 16

Learning Rate Schedule. We apply grid search for local and global learning rates and use cosine learning rate decay across local updates:

- **FedAvg**, **SCAFFOLD**, **FedCM**: local LR = `0.1`
- **FedLADA**, **Local AdamW**: local LR = `1e-4`
- **Local Muon**, **FMuon**: local LR = `1e-3`

Weight Decay. To ensure fair comparison under different regularization settings, we assign:

- **FedAvg**, **SCAFFOLD**, **FedCM**: weight decay = `0.001`
- **Local Muon**, **FedLADA**, **Local AdamW**, **FMuon**: weight decay = `0.01`

Other Settings. AdamW optimizers use $\beta_1 = 0.9$, $\beta_2 = 0.999$. Learning rates follow cosine decay without warmup.

OpenWebText Pre-Training. The GPT-2 experiments in the main text use 20 clients with 20% participation, 300 communication rounds, a local batch size of 16, and $K = 100$ local steps. These settings differ from the RoBERTa fine-tuning configurations above.

Table 12: Per-round communication cost of different momentum aggregation strategies. Here $|x|$ denotes the number of communicated model/update scalars, and $\mathrm{CommCost}$ is per-round communication cost, Compute-Cost is per-round computation time. (ViT-Tiny, $R = 300$, Dir-0.1, $K = 50$)

Method / Strategy	Communication	CommCost	Compute-Cost (s)	Acc(%)	Comm@23%Acc (MB)
FedAvg	$\|x\|$	22.8 MB	4.56 s	27.14	4190 MB
FedProx	$\|x\|$	22.8 MB	4.58 s	26.84	5244 MB
FedDyn	$\|x\|$	22.8 MB	4.75 s	27.66	4788 MB
Mime	$2\|x\|$	45.6 MB	5.26 s	27.76	9804 MB
FedAdam	$\|x\|$	22.8 MB	4.57 s	28.50	4651 MB
SCAFFOLD	$2\|x\|$	45.6 MB	5.22 s	27.31	8390 MB
FedCM	$\|x\|$	22.8 MB	4.68 s	31.18	6156 MB
FedLADA	$2\|x\|$	45.6 MB	5.02 s	31.50	4879 MB
Local AdamW	$\|x\|$	22.8 MB	4.89 s	37.86	1482 MB
FedMuon	$2\|x\|$	45.6 MB	5.56 s	45.17	1567 MB
Local Muon	$\|x\|$	22.8 MB	5.14 s	40.53	2394 MB
FMuon_SVD	$\|x\| + \|M\|_{\mathrm{SVD}}$	23.9 MB	5.25 s	47.68	550 MB

The reported compressed momentum payload adds approximately 5% to the baseline payload. The SVD storage ratio depends on matrix shape; see Section H.7.

Table 13: Per-round communication and computation cost of different methods on QQP using **RoBERTa-Base** with LoRA. Here $|x|$ denotes the number of communicated model/update scalars, $\mathrm{CommCost}$ is per-round communication cost (MB), and Compute-Cost is per-round computation time (s). (QQP, RoBERTa-Base, $R = 100$, Dir-0.5, 20 clients, 20% participation, batch size 16, $K = 50$)

Method / Strategy	**CommCost (MB)**	**Compute-Cost (s)**	**ACC(%)**
FedAvg	7.1 MB	8.56 s	81.75
FedProx	7.1 MB	8.62 s	81.15
FedDyn	7.1 MB	8.89 s	81.35
Mime	14.2 MB	11.25 s	80.26
FedAdam	7.1 MB	8.76 s	82.22
SCAFFOLD	14.2 MB	11.56 s	80.26
FedCM	7.1 MB	8.98 s	81.22
FedLADA	14.2 MB	9.58 s	81.56
Local AdamW	7.1 MB	9.26 s	81.51
FedMuon	14.2 MB	10.25 s	83.56
Local Muon	7.1 MB	9.73 s	83.06
FMuon_SVD (ours)	7.45 MB	9.91 s	84.65

H Additional Experiments

H.1 Communication and Computation Cost Analysis

As shown in Table 13, we evaluate various federated learning and local optimization strategies on the QQP dataset using RoBERTa-Base with LoRA for parameter-efficient fine-tuning. Most first-order methods (FedAvg, FedProx, FedDyn, FedAdam, FedCM, Local AdamW, Local Muon) require around 7.1 MB of communication per round, while methods that maintain additional control variables or gradient information (Mime, SCAFFOLD, FedLADA) incur a higher communication cost of 14.2 MB. In contrast, FMuon increases the per-round communication cost only slightly to 7.45 MB while achieving a notable improvement in model performance.

FMuon achieves 84.65% QQP accuracy, exceeding Local Muon by 1.59 percentage points. Its reported communication and computation costs increase from 7.1 to 7.45 MB and from 9.73 to 9.91 seconds, respectively, corresponding to approximately 4.9% and 1.8% overhead.

Table 14: Test accuracy, training loss of each method on CIFAR-100 using **ResNet-18** and **ViT-Tiny** over 300 communication rounds under Dir-0.6 and Dir-0.1 (100 clients, 10% participation, batch size 50, $K = 50$).

Method	**ResNet-18 (Dir-0.6)**		**ResNet-18 (Dir-0.1)**		**ViT-Tiny (Dir-0.6)**		**ViT-Tiny (Dir-0.1)**	
	Test Acc	Loss	Test Acc	Loss	Test Acc	Loss	Test Acc	Loss
FedAvg	64.08	0.376	60.25	0.767	32.36	2.350	27.14	2.867
FedProx	63.12	0.458	59.66	0.812	31.51	2.425	26.84	2.875
FedDyn	66.12	0.352	63.01	0.615	33.25	2.125	27.66	2.723
Mime	67.34	0.312	63.37	0.604	34.12	2.101	27.76	2.702
FedAdam	67.23	0.332	63.61	0.512	34.32	1.965	28.50	2.425
SCAFFOLD	65.01	0.365	62.56	0.814	32.17	2.295	27.31	2.855
FedCM	68.42	0.282	65.73	0.639	33.33	2.081	31.18	2.338
FedLADA	65.07	0.671	57.78	0.498	38.33	2.121	31.50	2.678
Local AdamW	62.84	0.363	58.97	0.794	40.47	1.026	37.86	1.954
`Local Muon`	71.66	0.395	66.71	1.504	46.69	0.201	40.53	1.432
`FMuon`	**74.12**	0.021	**73.05**	0.246	**50.22**	0.162	**48.18**	0.556

H.2 Additional Non-IID Baselines

To substantiate our method's advantages under non-IID conditions, we extend the comparison to additional federated baselines such as FedProx, FedDyn, and FedAdam, Mime. The comprehensive results are presented in Table 14.

Training on CIFAR-100 with ResNet-18. Table 14 present the test accuracy and training loss on CIFAR-100 using ResNet-18. `FMuon` achieves the best performance under both Dir-0.6 and Dir-0.1 settings, reaching a top accuracy of **74.12%** and **73.05%**, respectively. It also attains the lowest training loss (**0.021** and **0.246**), indicating improved final optimization performance. Compared to other adaptive baselines such as `Local AdamW`, `FMuon` shows superior generalization under data heterogeneity, confirming its effectiveness in CNNs.

Training on CIFAR-100 with ViT-Tiny. Table 14 shows `FMuon` achieves the best performance across both heterogeneity levels, with **50.22%** (Dir-0.6) and **48.18%** (Dir-0.1), and the lowest training loss (**0.162** and **0.556**), showing improved final optimization performance. Compared to `Local AdamW`, it provides consistent improvements in both final accuracy and training loss. Moreover, other adaptive baselines such as `FedLADA` perform significantly worse under high heterogeneity, highlighting the effectiveness of the proposed federated optimization procedure. These results validate that `FMuon` is particularly effective for federated vision Transformers under non-IID conditions. We additionally evaluate pretrained vision models in the main text.

H.3 IID and Non-IID Comparisons

As shown in Table 15, on CIFAR-100, our proposed `FMuon` consistently achieves the best test accuracy across both **ResNet-18** and **ViT-Tiny** under **IID** and **non-IID** (Dir-0.6 and Dir-0.1) settings. For ResNet-18, `FMuon` reaches 74.32 accuracy in the IID case, outperforming strong baselines such as Mime and FedAdam by approximately 6.4 and 6.2 percentage points, respectively. Even under highly heterogeneous data (Dir-0.1), `FMuon` still achieves 73.05, significantly higher than FedCM (65.73) and FedAdam (63.61). A similar trend is observed for ViT-Tiny: `FMuon` achieves 50.56 accuracy in the IID setting, nearly 10 percentage points higher than Local AdamW; and in the Dir-0.1 scenario, it maintains a strong performance of 48.18, outperforming all baselines by a large margin.

It is also noteworthy that `Local Muon` performs particularly well under IID conditions. For example, it achieves 72.04 with ResNet-18 and 47.69 with ViT-Tiny, indicating that `Local Muon` converges rapidly and achieves strong performance when the data distribution across clients is homogeneous. However, when the data becomes non-IID—especially under Dir-0.1—the performance of `Local Muon` drops significantly (e.g., from 72.04 to 66.71 on ResNet-18, and from 47.69 to 40.53 on ViT-Tiny), showing increased sensitivity to data heterogeneity. In contrast, our `FMuon` not only preserves the fast convergence and strong performance in the IID setting, but also effectively mitigates client drift under non-IID conditions. As a result, `FMuon` consistently achieves the best performance across

Table 15: Test accuracy of each method on CIFAR-100 using **ResNet-18** and **ViT-Tiny** over 300 communication rounds under IID, Dir-0.6 and Dir-0.1 (100 clients, 10% participation, batch size 50, $K = 50$).

Method	ResNet-18			ViT-Tiny		
	IID	Dir-0.6	Dir-0.1	IID	Dir-0.6	Dir-0.1
FedAvg	65.74	64.08	60.25	32.45	32.36	27.14
FedProx	65.52	63.12	59.66	32.25	31.51	26.84
SCAFFOLD	65.67	65.01	62.56	32.67	32.17	27.31
FedDyn	66.58	66.12	63.01	33.66	33.25	27.66
Mime	67.89	67.34	63.37	34.52	34.12	27.76
FedCM	70.57	68.42	65.73	27.88	33.33	31.18
FedLADA	66.23	65.07	57.78	38.56	38.33	31.50
FedAdam	68.12	67.23	63.61	34.83	34.32	28.50
Local AdamW	64.60	62.84	58.97	40.78	40.47	37.86
`Local Muon`	72.04	71.66	66.71	47.69	46.69	40.53
`FMuon`	**74.32**	**74.12**	**73.05**	**50.56**	**50.22**	**48.18**

Table 16: Test accuracy using RoBERTa-Base with LoRA across GLUE tasks over 100 communication rounds (Dir-0.8, 4 clients, 100% participation, batch size 16, $K = 50$).

Method (Dir-0.8)	CoLA	RTE	SST-2	QQP	MRPC	QNLI	MNLI
FedAvg	$57.32_{\pm 0.22}$	$62.71_{\pm 0.35}$	$93.32_{\pm 0.08}$	$84.13_{\pm 0.15}$	$87.02_{\pm 0.19}$	$90.19_{\pm 0.12}$	$84.18_{\pm 0.21}$
SCAFFOLD	$58.14_{\pm 0.25}$	$63.62_{\pm 0.28}$	$93.54_{\pm 0.09}$	$84.62_{\pm 0.17}$	$87.56_{\pm 0.22}$	$90.26_{\pm 0.11}$	$84.26_{\pm 0.20}$
FedCM	$58.14_{\pm 0.27}$	$66.14_{\pm 0.31}$	$93.61_{\pm 0.07}$	$84.56_{\pm 0.18}$	$87.11_{\pm 0.16}$	$90.08_{\pm 0.13}$	$84.32_{\pm 0.23}$
FedLADA	$59.10_{\pm 0.21}$	$74.14_{\pm 0.29}$	$93.66_{\pm 0.10}$	$84.86_{\pm 0.16}$	$87.42_{\pm 0.18}$	$90.18_{\pm 0.14}$	$84.46_{\pm 0.19}$
Local AdamW	$59.33_{\pm 0.23}$	$74.04_{\pm 0.27}$	$93.55_{\pm 0.11}$	$84.68_{\pm 0.15}$	$87.16_{\pm 0.20}$	$90.11_{\pm 0.12}$	$84.54_{\pm 0.18}$
`Local Muon`	$60.16_{\pm 0.20}$	$71.48_{\pm 0.34}$	$93.34_{\pm 0.09}$	$85.11_{\pm 0.13}$	$87.45_{\pm 0.21}$	$90.97_{\pm 0.15}$	$84.59_{\pm 0.17}$
`FMuon (ours)`	$\mathbf{63.04}_{\pm \mathbf{0.19}}$	$\mathbf{77.12}_{\pm \mathbf{0.30}}$	$\mathbf{94.12}_{\pm \mathbf{0.08}}$	$\mathbf{85.73}_{\pm \mathbf{0.14}}$	$\mathbf{88.23}_{\pm \mathbf{0.17}}$	$\mathbf{91.43}_{\pm \mathbf{0.10}}$	$\mathbf{85.05}_{\pm \mathbf{0.16}}$

all data distributions and model architectures, demonstrating its robustness and stability in federated learning.

H.4 Language-Model Fine-Tuning Results

Tables 16 and 17 report RoBERTa-Base fine-tuning results under Dir-0.8 with full participation and Dir-0.5 with partial participation, respectively. FMuon achieves the highest reported score on all seven tasks in both settings. Under Dir-0.5, its gains over the next-best method are 1.24 points on CoLA, 1.65 on RTE, 1.59 on QQP, and 0.58 on MNLI. The arithmetic mean of its seven reported task scores is 81.00; this is a descriptive mean of the listed scores, rather than a separate accuracy measurement. Since both the partition and participation settings differ, the comparison between these configurations does not isolate the effect of either factor.

Table 17: Test accuracy (%) using RoBERTa-Base with LoRA across GLUE tasks over 100 communication rounds under Dir-0.5 partition. (20 clients, 20% participation, batch size 16, $K = 50$)

Method (Dir-0.5)	CoLA	RTE	SST-2	QQP	MRPC	QNLI	MNLI
FedAvg	$51.00_{\pm 0.26}$	$51.99_{\pm 0.24}$	$93.04_{\pm 0.16}$	$81.75_{\pm 0.11}$	$88.24_{\pm 0.18}$	$89.36_{\pm 0.15}$	$81.72_{\pm 0.25}$
FedProx	$53.11_{\pm 0.14}$	$53.25_{\pm 0.21}$	$92.26_{\pm 0.18}$	$81.15_{\pm 0.11}$	$87.36_{\pm 0.12}$	$88.12_{\pm 0.14}$	$81.41_{\pm 0.21}$
FedDyn	$53.21_{\pm 0.28}$	$52.22_{\pm 0.30}$	$92.36_{\pm 0.21}$	$81.35_{\pm 0.21}$	$87.89_{\pm 0.11}$	$89.12_{\pm 0.21}$	$82.18_{\pm 0.21}$
Mime	$52.15_{\pm 0.17}$	$51.62_{\pm 0.21}$	$92.21_{\pm 0.28}$	$80.26_{\pm 0.18}$	$88.04_{\pm 0.12}$	$89.11_{\pm 0.21}$	$82.51_{\pm 0.20}$
FedAdam	$53.21_{\pm 0.28}$	$52.52_{\pm 0.31}$	$92.36_{\pm 0.25}$	$82.22_{\pm 0.28}$	$88.12_{\pm 0.34}$	$88.01_{\pm 0.23}$	$82.66_{\pm 0.22}$
SCAFFOLD	$52.15_{\pm 0.17}$	$50.65_{\pm 0.20}$	$93.28_{\pm 0.28}$	$80.26_{\pm 0.18}$	$88.35_{\pm 0.12}$	$89.32_{\pm 0.24}$	$82.11_{\pm 0.20}$
FedCM	$53.21_{\pm 0.28}$	$52.22_{\pm 0.30}$	$92.56_{\pm 0.25}$	$81.22_{\pm 0.28}$	$88.56_{\pm 0.13}$	$89.02_{\pm 0.23}$	$82.12_{\pm 0.27}$
FedLADA	$54.66_{\pm 0.17}$	$57.02_{\pm 0.08}$	$93.88_{\pm 0.16}$	$81.56_{\pm 0.20}$	$89.01_{\pm 0.28}$	$89.86_{\pm 0.29}$	$82.44_{\pm 0.17}$
Local AdamW	$55.38_{\pm 0.12}$	$59.57_{\pm 0.25}$	$93.81_{\pm 0.19}$	$81.51_{\pm 0.05}$	$88.73_{\pm 0.23}$	$89.55_{\pm 0.15}$	$82.86_{\pm 0.26}$
`Local Muon`	$55.54_{\pm 0.05}$	$64.93_{\pm 0.17}$	$93.58_{\pm 0.27}$	$83.06_{\pm 0.11}$	$88.95_{\pm 0.13}$	$90.52_{\pm 0.27}$	$84.63_{\pm 0.10}$
`FMuon(ours)`	$\mathbf{56.78}_{\pm \mathbf{0.11}}$	$\mathbf{66.58}_{\pm \mathbf{0.29}}$	$\mathbf{94.04}_{\pm \mathbf{0.25}}$	$\mathbf{84.65}_{\pm \mathbf{0.16}}$	$\mathbf{89.11}_{\pm \mathbf{0.07}}$	$\mathbf{90.64}_{\pm \mathbf{0.13}}$	$\mathbf{85.21}_{\pm \mathbf{0.18}}$

Table 18: Effect of local optimizer Muon on CIFAR-100 (Dir-0.1, 300 rounds). Numbers in parentheses denote absolute improvement over the baseline in the previous row.

Variant	**ResNet-18**	**ViT-Tiny**
Local SGD	60.25	27.14
Local Muon	66.71 (↑ 6.46)	40.53 (↑ 13.39)
SCAFFOLD	62.56	27.31
Local Muon + SCAFFOLD	68.23 (↑ 5.67)	42.26 (↑ 14.95)
FedCM	65.73	31.18
Local Muon + FedCM	68.77 (↑ 3.04)	44.23 (↑ 13.05)
FedDyn	63.01	27.66
Local Muon + FedDyn	67.15 (↑ 4.14)	43.56 (↑ 15.90)
FedProx	59.66	26.84
Local Muon + FedProx	67.25 (↑ 7.59)	43.11 (↑ 16.27)
Local SGD + $\bar{\boldsymbol{m}}+\boldsymbol{\Delta}_G$	66.28	32.56
Local Muon + $\bar{\boldsymbol{m}}+\boldsymbol{\Delta}_G$	**73.05** (↑ 6.77)	**48.18** (↑ 15.62)

H.5 Results across Model Scales

The model-scale experiment in the main text shows that Local AdamW improves from 37.86% on ViT-Tiny to 42.21% on ViT-Large. Local Muon improves from 40.53% to 46.54%, while FMuon improves from 48.18% to 56.24%. Thus, the FMuon–Local Muon gap increases from 7.65 to 9.70 percentage points. FMuon also exceeds the listed FedMuon baseline by 3.01–4.18 points across the four ViT sizes. These results support the effectiveness of FMuon across the evaluated model scales under the Dir-0.1 setting.

H.6 Effectiveness of Muon and our correction strategy.

Table 18 evaluates Muon within several federated algorithms. Replacing Local SGD with Local Muon improves accuracy by 6.46 and 13.39 percentage points on ResNet-18 and ViT-Tiny, respectively. Integrating Muon with SCAFFOLD, FedCM, FedDyn, or FedProx yields gains of 3.04–7.59 points on ResNet-18 and 13.05–16.27 points on ViT-Tiny. Combining Muon with our momentum aggregation and local–global alignment achieves the highest accuracy among the listed variants, 73.05% and 48.18%, respectively.

H.7 Relation to Existing Methods and Communication Accounting

FMuon combines shared momentum initialization with local–global alignment. Unlike a client-specific control variate, the alignment uses a shared global direction. The comparisons in Table 18 show that Muon also benefits existing federated algorithms; FMuon attains the highest accuracy among the evaluated combinations.

The cost tables report model/update payloads and auxiliary payloads under the stated accounting convention. They should not be interpreted as total network traffic across all clients and both communication directions without specifying how uploads, downloads, and broadcasts are counted. For a momentum matrix of size $m \times n$, a rank-q SVD representation requires $q(m + n + 1)$ scalars when both singular-vector factors and singular values are transmitted, compared with mn scalars for the dense matrix. Retaining 2.5% of singular components therefore gives an approximately 5% payload ratio for large square matrices; the ratio depends on matrix shape and is not universal. The reported costs correspond to approximately 4.8% additional payload on ViT-Tiny and 4.9% on RoBERTa-Base relative to the respective baseline payloads.

Table 19: Effect of different local optimizers on CIFAR-100 (Dir-0.1, 300 rounds). Numbers in parentheses denote absolute improvement over the baseline in the previous row.

Variant	ResNet-18	ViT-Tiny
Local SGD	60.25	27.14
Local SGD + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	66.28 (↑ 6.03)	32.56 (↑ 5.42)
Local AdamW	58.97	37.86
Local AdamW + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	64.25 (↑ 5.28)	41.26 (↑ 3.40)
Local Shampoo	59.62	37.56
Local Shampoo + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	66.52 (↑ 6.90)	42.25 (↑ 4.69)
Local Adafactor	58.23	36.52
Local Adafactor + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	65.52 (↑ 7.29)	40.11 (↑ 3.59)
Local LAMB	59.62	36.55
Local LAMB + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	64.35 (↑ 4.73)	38.65 (↑ 2.10)
FedLAMB	62.35	36.28
Local Muon	66.71	40.53
Local Muon + $\bar{\boldsymbol{m}}$+$\boldsymbol{\Delta}_G$	**73.05** (↑ 6.34)	**48.18** (↑ 7.65)

Table 20: Algorithmic comparison with the prior FedMuon method.

Aspect	Prior FedMuon	FMuon (ours)
Challenge	LMO aggregation bias under heterogeneous client data.	Local–global LMO direction mismatch and loss of momentum information across communication rounds.
Correction	Corrects momentum *before* the LMO: $\mathrm{lmo}(m_i - c_i + c)$.	Mixes directions *after* the LMO: $(1-\alpha)\,\mathrm{lmo}(m_i) - \alpha\Delta_G$.
Cross-round state	Maintains local momentum and control variates.	Shares aggregated momentum $\bar{M}$ to initialize participating clients; retains Δ_G at the server.
Communicated state	Model/update and control-variate payloads.	Model/update and shared-momentum payloads; the latter can be transmitted in low-rank form.
Additional computation	Control-variate correction before LMO evaluation.	Direction mixing after LMO evaluation; the compressed variant also computes an SVD.
Model scale evaluated	ResNet-18 in the cited experiments.	ResNet-18 (11.7M parameters), ViTs (22M–632M), and GPT-2 models (124M–1.5B).
Theory	Analyzes its bias-corrected LMO update.	Analyzes the proposed update with exact LMOs and uncompressed momentum; the main theorem assumes full participation.
Practical contribution	Provides an LMO-based federated optimization method with control-variate correction.	Evaluates shared-momentum Muon across vision and language models and studies low-rank momentum sharing to reduce auxiliary communication.

H.8 Impact of Δ_G and $\bar{m}$ on other optimizers

Table 19 evaluates momentum aggregation and local–global alignment with SGD, AdamW, Shampoo, Adafactor, LAMB, and Muon. The correction improves accuracy for every tested optimizer on both backbones. Muon gains 6.34 percentage points on ResNet-18 and 7.65 on ViT-Tiny, attaining the highest final accuracies, 73.05% and 48.18%, respectively. The ViT-Tiny gain is the largest among the tested optimizers; on ResNet-18, Adafactor and Shampoo gain more (7.29 and 6.90 points). These results support compatibility with the evaluated optimizers, without establishing universal improvements or convergence guarantees for all optimizers.