DRAM: MEMORY-EFFICIENT MUON VIA RANK REALLOCATION IN PRINCIPAL-RANDOM SUBSPACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Memory-efficient Muon methods aim to reduce storage overhead while preserving strong performance, yet existing approaches compromise either convergence guarantees or practical efficiency. In this paper, we focus on subspace optimization for Muon, where the choice of subspace determines the gradient information retained for updating. To preserve dominant information while providing the coverage needed for stochastic convergence guarantees, we construct a principal-random subspace that combines leading singular directions of the stochastic gradient with random directions sampled from their orthogonal complement. Under a fixed rank budget, there is a trade-off between principal exploitation and complementary coverage, making rank allocation important for practical optimization. Our key observation is that the preferred balance shifts from principal-heavy early in training to random-heavy later. Motivated by this shift, we propose DRAM, a memory-efficient Muon optimizer that reallocates principal and random ranks across stages while maintaining momentum and performing matrix-sign normalization entirely in the subspace. Theoretically, we establish convergence guarantees for general rank-allocation schedules with nondecreasing coverage factors, obtaining rates of $O(T^{-1/4})$ in the stochastic setting and $O(T^{-1/2})$ in the deterministic setting, with fixed allocation as a special case. Using a simple linear reallocation schedule, DRAM achieves the best low-memory performance across three pre-training scales and nine fine-tuning settings.

1 INTRODUCTION

Muon (Jordan et al., 2024) has emerged as an effective matrix-structured optimizer for large language model (LLM) training (Liu et al., 2025; Essential AI et al., 2025), with adoption in frontier LLMs including Kimi K3 (Kimi Team et al., 2026), GLM-5 (GLM-5-Team et al., 2026), DeepSeek-V4 (DeepSeek-AI et al., 2026), and Qwen3.8-Next (Qiu et al., 2026). For each matrix parameter, Muon maintains a dense momentum that aggregates historical gradients and applies matrix-sign normalization before the parameter update. As models scale, these full-size momentum states incur substantial memory overhead, motivating growing interest in memory-efficient variants of Muon.

Existing memory-efficient Muon methods have explored several routes, but remain limited in either theoretical guarantees or practical efficiency. Subspace-based approaches establish convergence guarantees either through probabilistic full-rank correction, which can incur high peak memory (Pan et al., 2025), or under reversible-layer assumptions that do not cover standard Transformer layers (Refael et al., 2025a). Momentum-compression methods reduce storage but still perform matrix-sign normalization on a full-dimensional momentum (Wang et al., 2026; Huang et al., 2026), while factorized methods require per-step factor updates (Mahdavinia & Mahdavi, 2025). This raises a natural question: *how can we design a memory-efficient Muon method that offers both practical efficiency and stochastic convergence guarantees for general matrix objectives?*

We pursue this goal through compact subspace optimization, which maintains momentum in a low-dimensional subspace via gradient projection. The choice of subspace is therefore critical, as it determines the gradient information retained for parameter updates. Under a first-order descent criterion, the ideal choice is the full-gradient TopSVD subspace, spanned by its leading singular directions. Since the full gradient is unavailable in stochastic optimization, a natural surrogate is stochastic-gradient TopSVD, which is used by GaLore (Zhao et al., 2024). However, optimization

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.

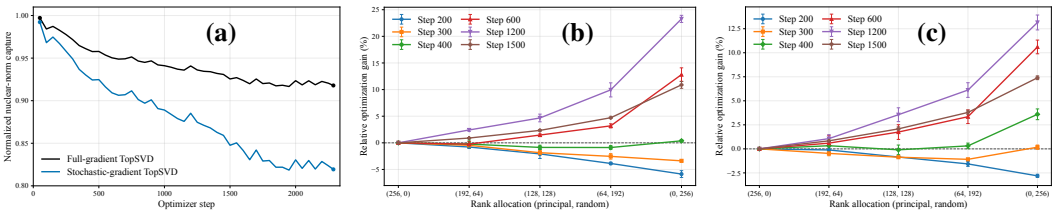


Figure 1: LLaMA-60M pre-training. (a) Rank-256 nuclear-norm capture by the full-gradient TopSVD and stochastic-gradient TopSVD along Full-Muon. (b)–(c) Relative gain in validation-loss reduction over the principal-only allocation along Full-Muon and GaLore-Muon trajectories.

over such a subspace can fail to converge under anisotropic gradient noise (He et al., 2025; Pan et al., 2025). GoLore (He et al., 2025) addresses this issue by replacing TopSVD with random subspaces, whose uniform full-space coverage in expectation enables it to establish stochastic convergence guarantees for Momentum SGD (MSGD). However, it may slow practical optimization since it does not utilize dominant gradient directions. To retain strong practical optimization while ensuring sufficient coverage, we propose a *principal-random subspace* that combines leading singular directions of the stochastic gradient with random directions sampled from their orthogonal complement.

The random directions uniformly cover the orthogonal complement of the principal subspace in expectation, which is a key property for our stochastic convergence analysis. A remaining question is how to allocate a given rank budget to achieve strong optimization performance in practice, since stronger complementary coverage necessarily leaves fewer dimensions for principal exploitation. Our key observation is that the preferred balance shifts over training. Fig. 1(a) shows that even full-gradient TopSVD captures a decreasing fraction of the full-space first-order descent signal, with stochastic-gradient TopSVD incurring an additional estimation gap. Figs. 1(b)–(c) further show that random directions are less favorable early but become beneficial later along both full-state Muon (*Full-Muon*) and the GaLore-style Muon (*GaLore-Muon*) trajectories. This shift motivates reallocating the rank budget from principal toward random directions over training.

Building on the principal-random subspace, we propose DRAM (**D**ynamic **R**ank **A**llocation for **M**uon), a memory-efficient Muon optimizer that reallocates rank between principal and random directions across stages while maintaining momentum and performing matrix-sign normalization entirely in the compact subspace. We then establish convergence guarantees under general rank-allocation schedules with nondecreasing coverage factors, obtaining rates of $O(T^{-1/4})$ in the stochastic setting and $O(T^{-1/2})$ in the deterministic setting over T updates. Table 1 compares DRAM with related Muon methods. With a linear reallocation schedule, DRAM consistently achieves strong performance relative to low-memory baselines across both pre-training and fine-tuning settings. Our contributions are summarized as follows:

- We introduce a *principal-random subspace* that preserves leading singular directions of the stochastic gradient while sampling random directions from their orthogonal complement, providing the coverage property for establishing stochastic convergence guarantees.
- We identify a stage-dependent shift in the preferred rank allocation for Muon from principal toward random directions over training, and propose DRAM, which supports reallocating the rank budget across optimization stages.
- We establish convergence guarantees for DRAM under general rank-allocation schedules with nondecreasing coverage factors, obtaining $O(T^{-1/4})$ and $O(T^{-1/2})$ rates in the stochastic and deterministic settings, respectively, with fixed allocation as a special case.
- Experiments show that DRAM achieves the highest average downstream accuracy among low-memory methods at three pre-training scales and the best result across nine fine-tuning settings, with ablations supporting the proposed mechanisms.

2 Related Work

We first review low-rank and subspace methods that reduce memory usage in LLM training, and then discuss memory-efficient Muon methods.

Table 1: Comparison of Muon methods for $X \in \mathbb{R}^{m\times n}$ with $m \leq n$. GaLore-Muon and GoLore-Muon use stochastic-gradient TopSVD and random subspaces, respectively. k denotes the subspace rank and p the full-rank layer-update probability in GUM. DRAM is shown with fixed total rank.

Method	Expected storage	Worst-case storage	Guarantee
Full-Muon	$O(mn)$	$O(mn)$	✓
GaLore-Muon / GoLore-Muon	$O((m+n)k)$	$O((m+n)k)$	–
GUM	$O((1-p)(m+n)k + p(mk+mn))$	$O(mn)$	✓
DRAM (Ours)	$O((m+n)k)$	$O((m+n)k)$	✓

2.1 Low-Rank and Subspace Methods

Parameter-efficient methods achieve memory efficiency by reducing the number of trainable parameters. LoRA (Hu et al., 2022) is a representative approach based on low-rank adapters, followed by extensions such as periodic adapter merging (Lialin et al., 2024) and magnitude-direction decomposition (Liu et al., 2024). We refer readers to Yang et al. (2025b) for a broader survey.

Full-parameter subspace methods reduce optimizer-state memory by projecting gradients into low-dimensional subspaces and maintaining states therein. GaLore (Zhao et al., 2024) constructs subspaces from the leading singular directions of stochastic gradients, whereas FLORA (Hao et al., 2024) and GoLore (He et al., 2025) use randomized projections. Recent methods reduce projection overhead through sparse projections (Muhamed et al., 2024), improve coverage through importance-weighted sampling (Zhang et al., 2025) or complementary direction selection (Gong et al., 2026), and adapt subspaces through gradient-rank reduction (Refael et al., 2025b), online PCA (Liang et al., 2024), Grassmannian tracking (Rajabi et al., 2025), and block power iteration (Robert et al., 2025). Other approaches exploit information outside projected states through norm-scaled residual updates (Chen et al., 2025a), low-rank estimation of gradient scaling (Zhu et al., 2025), or hybrid optimizers (Liu et al., 2026). Beyond gradient-projection methods, RSO (Chen et al., 2025b) formulates training as a sequence of low-dimensional subproblems over random subspaces.

Two Adam-based methods explore direction-selection mechanisms related to our design. Alice (Gong et al., 2026) periodically replaces part of a tracked eigenspace with directions sampled outside the previous subspace, while keeping the rank allocation fixed. AdaRankGrad (Refael et al., 2025b) adapts the rank of the gradient TopSVD subspace to exploit the decreasing effective gradient rank for memory savings, based on an information-retention threshold.

2.2 Memory-Efficient Muon

Muon (Jordan et al., 2024) has demonstrated strong performance in LLM training (Liu et al., 2025; Essential AI et al., 2025), with recent work analyzing its convergence properties (Li & Hong, 2025; Kim & Oh, 2026). Memory-efficient Muon variants reduce optimizer-state storage through low-dimensional or low-precision representations. GUM (Pan et al., 2025) retains GaLore-style projection and establishes stochastic convergence through probabilistic full-rank correction. However, layers selected for full-rank updates require full-size momentum states, resulting in the same worst-case storage as Full-Muon. SUMO (Refael et al., 2025a) performs moment orthogonalization within a dynamically adapted subspace, while its convergence guarantee is restricted to reversible layers and does not cover standard Transformer layers. MoFaSGD (Mahdavinia & Mahdavi, 2025) maintains a dynamically updated low-rank momentum factorization with per-step factor updates. LoRA-Pre (Wang et al., 2026) and LiMuon (Huang et al., 2026) compress persistent representations, but apply matrix-sign normalization to a full-dimensional momentum. Alternatively, 4-bit-Muon-GRASP (Wu et al., 2026) and MuonQ (Su et al., 2026) reduce storage through quantization, but introduce per-step decomposition or reconstruction. DRAM differs from these methods by maintaining momentum and performing matrix-sign normalization entirely in a low-dimensional subspace while establishing stochastic convergence for general nonconvex matrix objectives.

3 Method

In this section, we develop our method and establish convergence guarantees. For clarity, the presentation focuses on a single matrix, and the same construction is applied to each matrix parameter.

3.1 COMPACT MUON WITH PRINCIPAL-RANDOM SUBSPACES

We consider the stochastic matrix optimization problem $\min_{X\in\mathbb{R}^{m\times n}} F(X) := \mathbb{E}_{\xi\sim\mathcal{D}}[f(X;\xi)]$, where $\mathcal{D}$ denotes the data distribution and $f(X;\xi)$ is the stochastic loss. We assume $m \le n$ without loss of generality. When $m > n$, we apply the same construction to the transposed gradient. Given a stochastic gradient $G_t = \nabla f(X_t;\xi_t)$, Full-Muon performs updates according to

$$M_t = (1-\beta)M_{t-1} + \beta G_t, \qquad X_{t+1} = X_t - \eta\,\mathrm{msign}(M_t), \tag{1}$$

where $\beta \in (0,1]$ is the momentum coefficient, $\eta > 0$ is the stepsize, and $\mathrm{msign}(A) = UV^\top$ for a compact SVD $A = U\Sigma V^\top$. The full-size momentum $M_t \in \mathbb{R}^{m\times n}$ requires $O(mn)$ storage. To reduce this cost, we divide the T optimization steps into $S = \lceil T/\tau \rceil$ blocks of at most τ steps and maintain the momentum in a low-dimensional subspace that is fixed within each block. For block s, let $B_s \in \mathbb{R}^{m\times k_s}$ be an orthonormal basis with $1 \le k_s < m$. For $t > s\tau$ within block s, we update

$$H_t = (1-\beta)H_{t-1} + \beta B_s^\top G_t, \qquad X_{t+1} = X_t - \eta B_s\,\mathrm{msign}(H_t), \tag{2}$$

where $H_t \in \mathbb{R}^{k_s\times n}$ is the compact momentum. Compared with the full-space update in (1), (2) performs both momentum accumulation and normalization in the subspace, reducing optimizer-state storage and normalization cost. As shown in (2), the full-space gradient enters optimization only through its projection onto $\mathrm{Range}(B_s)$, making the construction of B_s a central design choice.

To derive a criterion for subspace selection, consider the momentum-free idealized Muon update $X_{t+1} = X_t - \eta B\,\mathrm{msign}(B^\top\nabla F(X_t))$. The objective decrease $\Delta_t = F(X_t) - F(X_{t+1})$ satisfies

$$\Delta_t = \eta\left\langle \nabla F(X_t), B\,\mathrm{msign}(B^\top\nabla F(X_t))\right\rangle_F + O(\eta^2) = \eta\|B^\top\nabla F(X_t)\|_* + O(\eta^2). \tag{3}$$

Thus, maximizing the first-order descent amounts to solving $\max_{B^\top B=I_k}\|B^\top\nabla F(X_t)\|_*$. By the Ky Fan k-norm variational characterization (Horn & Johnson, 2013), the ideal rank-k subspace is spanned by the leading k singular vectors of the full gradient $\nabla F(X_t)$, i.e., its TopSVD subspace.

In stochastic optimization, the full gradient is unavailable, and a natural surrogate is the TopSVD subspace of the stochastic gradient $G_t = \nabla f(X_t;\xi_t)$, which is adopted by GaLore (Zhao et al., 2024). However, under anisotropic gradient noise, such TopSVD-based subspace optimization can fail to converge with MSGD (He et al., 2025) or Muon (Pan et al., 2025), as also illustrated in Section 4.1. The underlying issue is that the greedy TopSVD selection can become dominated by gradient noise and fail to preserve true-gradient information. Motivated by GoLore (He et al., 2025), which uses random subspaces to provide full-space coverage, we retain the TopSVD directions and augment them with random directions sampled from their *orthogonal complement*.

Principal-random subspace. To exploit dominant stochastic-gradient components, at the beginning of block s, we compute $G_{s\tau} = U_{s\tau}\Sigma_{s\tau}V_{s\tau}^\top$ and retain the leading r_s left singular directions $R_s := U_{s\tau}[:,1:r_s] \in \mathbb{R}^{m\times r_s}$, termed *principal directions*. To provide coverage beyond the principal subspace, we sample q_s *random directions* uniformly from its orthogonal complement as

$$P_s \sim \mathrm{Unif}\{P\in\mathbb{R}^{m\times q_s} : P^\top P = I_{q_s},\ R_s^\top P = 0\}. \tag{4}$$

In practice, P_s is obtained by projecting an i.i.d. Gaussian matrix $Z_s \in \mathbb{R}^{m\times q_s}$ onto $\mathrm{Range}(I - R_sR_s^\top)$, followed by a sign-corrected thin QR factorization (Stewart, 1980; Mezzadri, 2007). The resulting basis $B_s = [R_s\ P_s] \in \mathbb{R}^{m\times(r_s+q_s)}$ spans the *principal-random subspace*, reducing to the GaLore-style principal subspace (Zhao et al., 2024) when $q_s = 0$ and the GoLore-style random subspace (He et al., 2025) when $r_s = 0$. We refer to the corresponding Muon baselines as *GaLore-Muon* and *GoLore-Muon*, respectively, and present their update rules in Appendix A.1. The following lemma characterizes the coverage provided by the principal-random subspace.

Lemma 1 *Let $R \in \mathbb{R}^{m\times r}$ satisfy $R^\top R = I_r$, and define $\Pi = RR^\top$ and $Q = I - \Pi$. Let $P \in \mathbb{R}^{m\times q}$ be sampled as in (4) with $0 < q \le m - r$, and let $B = [R\ P]$ and $\delta = q/(m-r)$. Then $\mathbb{E}[BB^\top \mid R] = \Pi + \delta Q$ and $\mathbb{E}\big[\|B^\top A\|_* \mid R\big] \ge \|(\Pi+\delta Q)A\|_* \ge \delta\|A\|_*$ for any fixed $A \in \mathbb{R}^{m\times n}$.*

Lemma 1 shows that the principal subspace is fully retained, while its orthogonal complement is uniformly covered in expectation with strength δ. Importantly, at block starts, the nuclear-norm bound links the projected nuclear norm $\|B^\top\nabla F(X_t)\|_*$ in the first-order descent of subspace Muon to the full-space stationarity measure $\|\nabla F(X_t)\|_*$, controlling the latter in expectation up to a factor $1/\delta$. This connection is key to establishing the stochastic convergence guarantees in Section 3.3.

Algorithm 1 Single-Matrix DRAM

Input: $X_0 \in \mathbb{R}^{m\times n}$ with $m \le n$, data distribution $\mathcal{D}$, target horizon T, stepsize η, momentum parameter $\beta \in (0,1]$, block length τ, and schedule $\{(r_s, q_s)\}_{s=0}^{S-1}$ with $S = \lceil T/\tau \rceil$

for $s = 0, 1, \ldots, S-1$ **do**
  for $t = s\tau, s\tau+1, \ldots, \min\{(s+1)\tau, T\} - 1$ **do**
    Sample $\xi_t \sim \mathcal{D}$ and compute $G_t = \nabla f(X_t; \xi_t)$
    if $t = s\tau$ **then**
      Compute $G_t = U_t \Sigma_t V_t^\top$ and set $R_s = U_t[:, 1:r_s]$
      Sample P_s according to (4) and set $B_s = [R_s \ P_s]$
    end if
    Compute $\widehat{G}_t = B_s^\top G_t$ and update H_t by (5)
    Update $X_{t+1} = X_t - \eta B_s \operatorname{msign}(H_t)$
  end for
end for
return X_T.

Subspace construction cost. For $G \in \mathbb{R}^{m\times n}$ with $m \le n$, approximating its leading r_s left singular vectors via randomized SVD costs $O(mnr_s)$ (Halko et al., 2011). Given R_s, generating P_s requires $O(mr_sq_s)$ operations for complementary projection and $O(mq_s^2)$ for thin QR factorization. Thus, constructing B_s costs $O(mnr_s + mq_sk_s)$, where $k_s = r_s + q_s$. For a fixed total rank k_s, this cost interpolates between $O(mk_s^2)$ for GoLore-Muon and $O(mnk_s)$ for GaLore-Muon. Moreover, the construction cost is amortized across the τ steps in each block.

3.2 Dynamic Rank Allocation

Under a fixed total rank, allocating more rank to random directions strengthens coverage but leaves less for principal directions, creating a trade-off between principal exploitation and complementary coverage. We next examine how this balance shifts over training and design the allocation strategy.

Recall from (3) that $\|B^\top \nabla F(X_t)\|_*$ characterizes the first-order descent captured by the subspace spanned by B. We first compare rank-256 stochastic-gradient and full-gradient TopSVD by their nuclear-norm capture relative to $\|\nabla F(X_t)\|_*$. Fig. 1(a) shows decreasing normalized nuclear-norm capture even for full-gradient TopSVD, with stochastic-gradient TopSVD exhibiting a widening estimation gap. Following GoLore's observation that late-stage switching from TopSVD to random projections can improve AdamW and MSGD (He et al., 2025), we examine whether random directions benefit Muon over training. At representative checkpoints, we continue training for 200 steps with fixed total rank 256 while varying the rank allocation. Figs. 1(b)–(c) reveal a clear stage-dependent shift, with random directions being unfavorable early but beneficial later along both Full-Muon and GaLore-Muon trajectories. More details are provided in Section 4.2. These results suggest that the preferred allocation should vary across training stages.

Motivated by the preference shift, we propose DRAM (**D**ynamic **R**ank **A**llocation for **M**uon), which dynamically reallocates the rank budget between principal and random directions across stages. We next specify the schedule and the corresponding compact Muon update.

Rank allocation. DRAM supports a general blockwise rank-allocation schedule $\{(r_s, q_s)\}_{s=0}^{S-1}$. Consistent with the observed shift toward random directions, we focus on schedules with nondecreasing coverage factor $\delta_s = q_s/(m - r_s)$, while allowing $k_s = r_s + q_s$ to vary across blocks. This includes both gradual reallocation and hard switching, which we empirically compare in Section 4.5. Unless otherwise specified, we fix $k_s \equiv k$ and use the linear schedule $q_s = \max\{\lfloor sk/(S-1) \rfloor, 1\}$ with $r_s = k - q_s$, progressively reallocating (r_s, q_s) from $(k-1, 1)$ to $(0, k)$ over training.

DRAM update. At the beginning of block s, DRAM constructs $B_s \in \mathbb{R}^{m\times k_s}$ based on the scheduled allocation (r_s, q_s) as in Section 3.1. For each stochastic gradient $G_t = \nabla f(X_t; \xi_t)$, DRAM projects it into the subspace as $\widehat{G}_t = B_s^\top G_t \in \mathbb{R}^{k_s\times n}$, and updates the compact momentum by

$$H_t = \begin{cases} \beta \widehat{G}_t, & t = s\tau, \\ (1-\beta) H_{t-1} + \beta \widehat{G}_t, & t > s\tau. \end{cases} \tag{5}$$

DRAM then applies the matrix sign to H_t in the compact subspace and lifts it back to the full space to update $X_{t+1} = X_t - \eta B_s \operatorname{msign}(H_t)$. The complete procedure is summarized in Algorithm 1.

Memory and normalization cost. In each block s, DRAM stores the projection basis $B_s \in \mathbb{R}^{m\times k_s}$ and compact momentum $H_t \in \mathbb{R}^{k_s\times n}$, requiring $O((m+n)k_s)$ persistent storage. With the same rank, GaLore-Muon and GoLore-Muon have the same order of storage cost. GUM (Pan et al., 2025) also maintains compact states for low-rank updates, but a layer sampled for full-rank correction stores an $m \times k$ projection matrix and an $m \times n$ momentum, yielding $O(mn)$ worst-case storage. Moreover, DRAM computes the matrix-sign normalization directly on H_t, reducing the cost of each Newton–Schulz iteration from $O(m^2 n)$ in the full space to $O(k_s^2 n)$ in the compact subspace.

3.3 Convergence Analysis

Following prior work on Muon (Pan et al., 2025; Kim & Oh, 2026), we analyze DRAM under the operator/nuclear-norm geometry and impose the following standard assumptions.

Assumption 1 *The objective function is lower bounded, i.e., $F^* := \inf_X F(X) > -\infty$.*

Assumption 2 *Let $\{\mathcal{F}_t\}_{t\ge 0}$ be the natural filtration generated by all randomness before sampling ξ_t. For every t, $\mathbb{E}[\nabla f(X_t;\xi_t) \mid \mathcal{F}_t] = \nabla F(X_t)$ and $\mathbb{E}[\|\nabla f(X_t;\xi_t) - \nabla F(X_t)\|_F^2 \mid \mathcal{F}_t] \le \sigma^2$.*

Assumption 3 *The objective $F : \mathbb{R}^{m\times n} \to \mathbb{R}$ is L_*-smooth with respect to the operator/nuclear norm pair, namely, for any $X, Y \in \mathbb{R}^{m\times n}$, $\|\nabla F(X) - \nabla F(Y)\|_* \le L_* \|X - Y\|_{\mathbf{op}}$.*

Assumption 4 *For the theoretical analysis, we represent Muon normalization by the exact matrix sign map $\operatorname{msign}(A) = UV^\top$ for an SVD $A = U\Sigma V^\top$.*

Under stochastic gradients, the complementary coverage factor $\delta_s = q_s/(m - r_s)$ from Lemma 1 relates subspace descent to full-gradient stationarity. In the deterministic setting, we obtain the full gradient and the coverage factor becomes $\gamma_s = k_s/m$ with $k_s = r_s + q_s$. In this case, random directions are unnecessary and q_s can be set to zero. We establish convergence guarantees for stage-dependent rank-allocation schedules with nondecreasing coverage factors as follows.

Theorem 1 *Suppose Assumptions 1–4 hold and let $\Delta := F(X_0) - F^* > 0$. Choose parameters specified in Appendix B, and then prescribe a schedule $\{(r_s, q_s)\}_{s=0}^{S-1}$ independently of the optimization randomness. For $\sigma > 0$, fix a rank cap $1 \le k_{\max} \le m$, and assume $0 \le r_s < m$, $1 \le q_s \le m - r_s$, $k_s = r_s + q_s \le k_{\max}$, and $\{\delta_s\}_{s=0}^{S-1}$ is nondecreasing. For $\sigma = 0$, assume $0 \le r_s \le m$, $0 \le q_s \le m - r_s$, $k_s > 0$, and $\{\gamma_s\}_{s=0}^{S-1}$ is nondecreasing. Then, Algorithm 1 satisfies*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(X_t)\|_* = \begin{cases} O\left(\Gamma_\delta \left[\sqrt{\frac{L_*\Delta}{T}} + \left(\frac{k_{\max} L_* \Delta \sigma^2}{T}\right)^{1/4}\right]\right), & \sigma > 0, \\ O\left(\Gamma_\gamma \sqrt{\frac{L_*\Delta}{T}}\right), & \sigma = 0. \end{cases}$$

where $\Gamma_\delta := \delta_0^{-1} + \frac{1}{S}\sum_{s=0}^{S-1} \delta_s^{-1}$ and $\Gamma_\gamma := \gamma_0^{-1} + \frac{1}{S}\sum_{s=0}^{S-1} \gamma_s^{-1}$.

Corollary 1 *Suppose the conditions of Theorem 1 hold with $r_s \equiv r$ and $q_s \equiv q$, and let $k = r + q$. For $\sigma > 0$, $\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(X_t)\|_* = O(\delta^{-1}[\sqrt{L_*\Delta/T} + (L_*\Delta k\sigma^2/T)^{1/4}])$ with $\delta = q/(m-r)$. For $\sigma = 0$ and $k > 0$, $\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(X_t)\|_* = O(m\sqrt{L_*\Delta}/(k\sqrt{T}))$.*

Remark. Theorem 1 establishes $O(T^{-1/4})$ stochastic and $O(T^{-1/2})$ deterministic convergence rates up to rank-schedule factors, matching the corresponding orders for Muon (Kim & Oh, 2026). To the best of our knowledge, this provides the first stochastic convergence guarantee for a memory-efficient Muon method that keeps both momentum accumulation and normalization in a compact subspace and applies to general matrix objectives.

4 Experiments

We evaluate DRAM through synthetic optimization, stage-wise rank-allocation analysis, LLM pre-training, full-parameter fine-tuning, and rank-allocation schedule ablations.

4.1 SYNTHETIC SETTINGS

Following prior work (He et al., 2025; Pan et al., 2025), we consider a quadratic problem with anisotropic gradient noise:

$$\min_{X\in\mathbb{R}^{n\times n}} F(X) = \frac{1}{2}\|AX\|_F^2 + \langle B, X\rangle_F, \qquad \nabla f(X;\xi) = A^\top AX + B + \xi\sigma C, \tag{6}$$

where $A = [I_{n-h}, 0] \in \mathbb{R}^{(n-h)\times n}$, $B = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{n\times n}$ and $C = \begin{bmatrix} 0 & 0 \\ 0 & I_h \end{bmatrix} \in \mathbb{R}^{n\times n}$. Here $D \in \mathbb{R}^{(n-h)\times(n-h)}$ has i.i.d. standard Gaussian entries, $\xi \sim \mathrm{Unif}\{-1, 1\}$, and σ controls the noise level. We set $n = 20$, $h = 12$, and $\sigma = 1$. The true gradient and noise lie in orthogonal subspaces.

Algorithms. We compare the fixed-allocation variant DRAM-F with Full-Muon, GaLore-Muon, GoLore-Muon, and GUM. All methods run for 3000 iterations with a learning rate of 5×10^{-3}. Subspace methods refresh their projections every 100 iterations. GaLore-Muon and GoLore-Muon use rank 8, and DRAM-F uses $(r, q) = (6, 2)$. GUM uses rank 4 and full-rank correction probability $p = 0.5$ to match the expected storage cost.

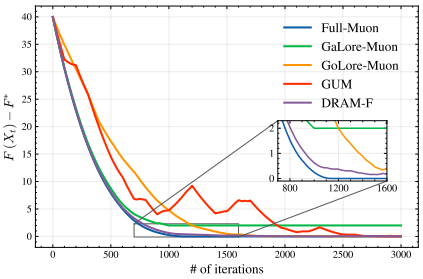


Figure 2: Synthetic optimization.

Results. Fig. 2 plots $F(X_t) - F^*$, where $F^* = -\frac{1}{2}\|D\|_F^2$ is the optimum. Full-Muon converges rapidly and serves as a reference trajectory, while the low-memory baselines exhibit different limitations. GaLore-Muon decreases the error quickly at first but plateaus at a nonzero gap. GoLore-Muon converges more slowly, whereas GUM shows a more oscillatory trajectory. In contrast, DRAM-F converges stably at a speed comparable to Full-Muon and outperforms the other low-memory methods, demonstrating the benefit of combining principal directions for fast optimization with complementary coverage to maintain descent under anisotropic noise.

4.2 STAGE-WISE RANK ALLOCATION ANALYSIS

Along a Full-Muon LLaMA-60M pre-training trajectory on C4 (Raffel et al., 2020), we measure normalized nuclear-norm capture $C_t(B) = \sum_\ell \|B_{t,\ell}^\top \bar{G}_{t,\ell}\|_* / \sum_\ell \|\bar{G}_{t,\ell}\|_*$, where ℓ indexes all matrix parameters, with full gradients $\bar{G}_{t,\ell}$ and projection bases $B_{t,\ell}$. Under the idealized Muon update in (3), $C_t(B)$ measures the fraction of full-space first-order descent retained by subspace optimization. We compare rank-256 full-gradient and stochastic-gradient TopSVD, averaging the latter over 64 independent mini-batches. Fig. 1(a) shows that their capture drops from 0.997 to 0.918 and from 0.992 to 0.819, respectively, revealing reduced fixed-rank capture and an increased stochastic estimation gap. To test rank-allocation preferences, we run 200-step continuations from Full-Muon and GaLore-Muon checkpoints with $r + q = 256$ and $q \in \{0, 64, 128, 192, 256\}$. Relative gain over the principal-only baseline is $\mathcal{G}_t(r, q) = (L_t^{(256,0)}(200) - L_t^{(r,q)}(200))/(L_t(0) - L_t^{(256,0)}(200)) \times 100\%$, where $L_t(0)$ and $L_t^{(r,q)}(200)$ are validation losses before and after continuation. Using this metric, Figs. 1(b)–(c) show that allocating more rank to random directions is unfavorable early but becomes beneficial later, with the mean $\pm$ one standard deviation reported over three seeds. This stage-dependent shift supports progressive rank reallocation from principal to random directions. Results for total ranks 64 and 128 are provided in Appendix A.2.

4.3 LLM PRE-TRAINING

Setup. We pre-train LLaMA-60M, 130M, and 350M from scratch on C4 (Raffel et al., 2020) for 1.2B, 2.5B, and 7B tokens, respectively. Following Pan et al. (2025), we evaluate zero-shot accuracy on ARC (Clark et al., 2018), OpenBookQA (Mihaylov et al., 2018), HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020), SIQA (Sap et al., 2019), and WinoGrande (Sakaguchi et al., 2021). We compare DRAM with AdamW and Full-Muon as full-memory references and with GaLore-Muon, GoLore-Muon, and GUM as low-memory baselines. GaLore-Muon and GoLore-Muon use rank 256. DRAM-F fixes $(r, q) = (128, 128)$, while DRAM linearly reallocates (r, q) from $(255, 1)$ to $(0, 256)$ over training. GUM uses rank 128, with the full-rank correction probability chosen to

Table 2: Zero-shot downstream accuracy (%) of models pre-trained on C4. Dashed lines separate full-memory and low-memory methods. In the Avg. column, the best and second-best results among low-memory methods are highlighted in **bold** and underlined, respectively.

Model	Method	ARC-E	ARC-C	OBQA	HellaSwag	PIQA	SIQA	WinoGrande	Avg.
LLaMA-60M	AdamW	29.67	21.50	28.20	26.38	58.49	35.72	50.67	35.80
	Full-Muon	32.37	22.10	27.00	26.33	59.52	35.52	52.49	36.47
	GaLore-Muon	31.19	20.48	26.20	26.02	58.87	35.77	53.04	35.94
	GoLore-Muon	31.10	19.97	28.00	26.16	59.52	35.52	51.62	35.98
	GUM	31.14	20.22	27.80	25.93	58.76	33.78	52.49	35.73
	DRAM-F	31.48	20.48	28.40	25.98	59.30	34.54	51.70	35.98
	DRAM	30.81	21.50	28.80	26.04	58.98	35.06	52.80	**36.28**
LLaMA-130M	AdamW	34.34	21.84	28.20	28.30	61.37	36.18	52.41	37.52
	Full-Muon	34.13	22.10	27.60	28.82	62.24	36.08	50.28	37.32
	GaLore-Muon	33.21	21.84	27.60	27.16	60.94	35.16	51.85	36.82
	GoLore-Muon	33.54	20.73	29.80	27.07	59.90	35.36	50.99	36.77
	GUM	33.84	21.84	28.20	27.36	60.12	35.88	52.57	37.11
	DRAM-F	33.29	21.67	27.80	27.19	61.37	36.23	51.62	37.02
	DRAM	33.12	22.61	29.00	26.94	60.39	34.80	53.43	**37.19**
LLaMA-350M	AdamW	36.95	22.70	27.80	33.83	66.76	37.67	53.35	39.87
	Full-Muon	37.92	23.38	27.40	34.88	66.92	38.64	52.33	40.21
	GaLore-Muon	34.72	22.61	27.20	30.49	63.93	36.18	50.36	37.93
	GoLore-Muon	36.03	22.61	26.00	30.33	63.22	36.64	52.33	38.17
	GUM	36.20	22.10	27.20	30.80	63.87	36.34	52.64	38.45
	DRAM-F	35.44	22.78	28.20	30.42	63.76	35.77	51.22	38.23
	DRAM	34.93	23.89	28.20	30.45	63.55	37.31	51.54	**38.55**

match the expected storage of the other low-memory methods. All Muon-based methods are tuned over the same scale-specific learning-rate grids. More details are provided in Appendix A.3.

Results. Table 2 shows that DRAM achieves the highest average downstream accuracy among low-memory methods at all three model scales. With a fixed allocation, DRAM-F matches or outperforms both GaLore-Muon and GoLore-Muon in average downstream accuracy at all three scales, suggesting that combining principal and random directions can be beneficial relative to either endpoint. Moreover, DRAM consistently improves over DRAM-F across the three scales, supporting the benefit of stage-wise rank reallocation. Overall, these results demonstrate the effectiveness of both the principal-random subspace design and stage-wise rank reallocation during pre-training.

4.4 LLM Fine-tuning

Setup. We perform full-parameter supervised fine-tuning of Qwen3-8B (Yang et al., 2025a), LLaMA-3.1-8B (Grattafiori et al., 2024), and Gemma-2-9B (Gemma Team et al., 2024) on instruction following, mathematical reasoning, and code generation. We fine-tune each model for one epoch on GPT4-LLM-Cleaned (Peng et al., 2023), GSM8K (Cobbe et al., 2021), and MBPP (Austin et al., 2021), and evaluate on IFEval (Zhou et al., 2023) with strict prompt-level accuracy, GSM8K with accuracy, and MBPP with Pass@1. We compare DRAM with Full-Muon, GaLore-Muon, GoLore-Muon, and GUM. DRAM, GaLore-Muon, and GoLore-Muon use rank 512, while GUM uses rank 128 with a model-specific full-rank correction probability set to match the expected storage of these low-memory methods. Additional details are provided in Appendix A.4.

Results. Table 3 shows the practical memory savings on MBPP, where DRAM reduces peak GPU memory by 19.6–20.7% relative to Full-Muon across the three backbones. The complete results in Appendix A.5 show reductions of 17.9–20.7% across all nine settings. Despite this memory reduction, Table 4 shows that DRAM achieves the best low-memory result in eight of the nine settings and ties for the best in the remaining one. It also matches or exceeds Full-Muon in eight of nine settings.

Table 3: Peak GPU memory (GiB) on MBPP.

Backbone	Full → DRAM	Reduction
Qwen3-8B	52.80 → 42.47	19.6%
LLaMA-3.1-8B	50.81 → 40.31	20.7%
Gemma-2-9B	60.25 → 48.06	20.2%

Table 4: Fine-tuning performance (%) across three tasks and three model backbones. Dashed lines separate full-memory and low-memory methods. The best and second-best results among low-memory methods are highlighted in **bold** and underlined, respectively.

Model	Method	Instruction following	Math	Code generation
Qwen3-8B	Full-Muon	50.46	84.16	71.60
	GaLore-Muon	50.28	83.47	71.80
	GoLore-Muon	47.87	83.47	72.00
	GUM	48.98	83.62	71.80
	DRAM	**50.46**	**84.16**	**72.20**
LLaMA-3.1-8B	Full-Muon	31.79	52.77	49.40
	GaLore-Muon	30.87	49.89	49.40
	GoLore-Muon	30.68	48.67	49.20
	GUM	30.13	50.27	**50.20**
	DRAM	**31.79**	**51.02**	**50.20**
Gemma-2-9B	Full-Muon	38.45	75.06	52.40
	GaLore-Muon	38.82	74.68	54.00
	GoLore-Muon	39.00	73.31	54.20
	GUM	39.37	74.83	54.40
	DRAM	**40.11**	**75.21**	**55.20**

Table 5: Rank-schedule ablation on pre-training validation loss. $\Delta L_{\text{val}} = L_{\text{Principal}} - L_{\text{schedule}}$, where larger values indicate greater improvement over Principal-only. We report the mean $\pm$ standard deviation over three random seeds. Realloc. denotes reallocation; HP denotes hyperparameter.

	Schedule properties			ΔL_{val} ($\times 10^{-3}$) $\uparrow$	
Schedule	Coverage	Rank realloc.	No schedule HP	LLaMA-60M	LLaMA-130M
Principal-only	–	–	✓	0.0 (ref.)	0.0 (ref.)
Random-only	✓	–	✓	-7.7 ± 1.4	-8.9 ± 2.7
Fixed $(128, 128)$	✓	–	✓	$+0.1 \pm 1.4$	-2.2 ± 0.4
Switch-25%	✓	✓	–	$+1.6 \pm 0.4$	$+1.4 \pm 0.3$
Switch-50%	✓	✓	–	$+0.5 \pm 0.2$	0.0 ± 0.2
Switch-75%	✓	✓	–	0.0 ± 0.1	-0.3 ± 0.03
Linear (Default)	✓	✓	✓	$+1.2 \pm 1.0$	$+0.9 \pm 0.05$

4.5 Ablation on Rank-Allocation Schedules

Setup. We fix the total rank at 256 and compare Principal-only, Random-only, a fixed $(128, 128)$ allocation, hard-switch schedules, and the default Linear schedule. Each hard-switch schedule starts from $(255, 1)$ and switches directly to $(0, 256)$ at 25%, 50%, or 75% of training, while Linear gradually reallocates rank between the same endpoints. Table 5 reports the validation-loss improvement over Principal-only, averaged over three random seeds.

Results. First, Random-only yields higher validation loss than Principal-only on both model sizes, showing that coverage alone is insufficient in these settings. Second, the fixed $(128, 128)$ allocation provides little benefit on LLaMA-60M and degrades LLaMA-130M, suggesting that a static allocation is not consistently effective. Finally, hard-switch performance is sensitive to the switch point, whereas the Linear schedule reduces validation loss for both model sizes without an additional switch-point hyperparameter, supporting its use as our default for stage-wise rank reallocation.

5 Conclusion

We introduce DRAM, a memory-efficient Muon optimizer built on a principal-random subspace with stage-wise rank reallocation between principal and random directions. This design achieves strong practical performance with provable convergence, while keeping momentum and normalization in a compact subspace. We establish $O(T^{-1/4})$ stochastic and $O(T^{-1/2})$ deterministic convergence rates and demonstrate strong performance across pre-training and fine-tuning.

AI USE STATEMENT

We used generative AI tools only for manuscript editing to improve readability, grammar, and style. All AI-assisted edits were reviewed by the authors for accuracy and consistency with the intended technical content. We take responsibility for the final content of this work.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *ArXiv e-prints*, arXiv:2108.07732, 2021.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020.

Xi Chen, Kaituo Feng, Changsheng Li, Xunhao Lai, Xiangyu Yue, Ye Yuan, and Guoren Wang. Fira: Can we achieve full-rank training of llms under low-rank constraint? In *Advances in Neural Information Processing Systems 38*, pp. 120680–120712, 2025a.

Yiming Chen, Yuan Zhang, Yin Liu, Kun Yuan, and Zaiwen Wen. A memory efficient randomized subspace optimization method for training large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 9747–9763, 2025b.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv e-prints*, arXiv:1803.05457, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *ArXiv e-prints*, arXiv:2110.14168, 2021.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *ArXiv e-prints*, arXiv:2606.19348, 2026.

Essential AI, Ishaan Shah, Anthony M. Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, et al. Practical efficiency of muon for pretraining. *ArXiv e-prints*, arXiv:2505.02222, 2025.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *ArXiv e-prints*, arXiv:2408.00118, 2024.

GLM-5-Team, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, et al. Glm-5: from vibe coding to agentic engineering. *ArXiv e-prints*, arXiv:2602.15763, 2026.

Wenbo Gong, Meyer Scetbon, Chao Ma, and Edward Meeds. Towards efficient optimizer design for LLM via structured fisher approximation with a low-rank extension. In *The Fourteenth International Conference on Learning Representations*, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *ArXiv e-prints*, arXiv:2407.21783, 2024.

N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

Yongchang Hao, Yanshuai Cao, and Lili Mou. Flora: Low-rank adapters are secretly gradient compressors. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 17554–17571, 2024.

Yutong He, Pengrui Li, Yipeng Hu, Chuyan Chen, and Kun Yuan. Subspace optimization for large language models with convergence guarantees. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 22468–22522, 2025.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2013.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Feihu Huang, Yuning Luo, and Songcan Chen. Limuon: Light and fast muon optimizer for large models. In *Forty-third International Conference on Machine Learning*, 2026.

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

Gyu Yeol Kim and Min-hwan Oh. Convergence of muon with newton-schulz. In *The Fourteenth International Conference on Learning Representations*, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, M. C., Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, et al. Kimi k3: Open frontier intelligence. *ArXiv e-prints*, arXiv:2607.24653, 2026.

Jiaxiang Li and Mingyi Hong. A note on the convergence of muon. *ArXiv e-prints*, arXiv:2502.02900, 2025.

Vladislav Lialin, Sherin Muckatira, Namrata Shivagunde, and Anna Rumshisky. ReloRA: High-rank training through low-rank updates. In *The Twelfth International Conference on Learning Representations*, 2024.

Kaizhao Liang, Bo Liu, Lizhang Chen, and Qiang Liu. Memory-efficient llm training with online subspace descent. In *Advances in Neural Information Processing Systems 37*, pp. 64412–64432, 2024.

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 e-prints*, arXiv:2502.16982, 2025.

Liming Liu, Zhenghao Xu, Zixuan Zhang, Hao Kang, Zichong Li, Chen Liang, Weizhu Chen, and Tuo Zhao. COSMOS: A hybrid adaptive optimizer for efficient training of large language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 32100–32121, 2024.

Pouria Mahdavinia and Mehrdad Mahdavi. Low-rank momentum factorization for memory efficient training. *Transactions on Machine Learning Research*, 2025.

Francesco Mezzadri. How to generate random matrices from the classical compact groups. *Notices of the American Mathematical Society*, 54(5):592–604, 2007.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, 2018.

Aashiq Muhamed, Oscar Li, David Woodruff, Mona T. Diab, and Virginia Smith. GRASS: Compute efficient low-memory LLM training with structured sparse gradients. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 14978–15003, 2024.

Rui Pan, Yang Luo, Yuxing Liu, Yang You, and Tong Zhang. Unbiased gradient low-rank projection. *ArXiv e-prints*, arXiv:2510.17802, 2025.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *ArXiv e-prints*, arXiv:2304.03277, 2023.

Zihan Qiu, Zekun Wang, Xiao Li, Yanpeng Li, Yang Xu, Yixuan Wang, Huaqing Zhang, Rui Men, Bochao Mao, Chengruidong Zhang, et al. On the design of qwen3.8-next architecture: Evaluation, efficiency, and training stability. *ArXiv e-prints*, arXiv:2608.30320, 2026.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Sahar Rajabi, Nayeema Nonta, and Sirisha Rambhatla. Subtrack++: Gradient subspace tracking for scalable llm training. In *Advances in Neural Information Processing Systems 38*, pp. 31584–31613, 2025.

Yehonathan Refael, Guy Smorodinsky, Tom Tirer, and Ofir Lindenbaum. Sumo: Subspace-aware moment-orthogonalization for accelerating memory-efficient llm training. In *Advances in Neural Information Processing Systems 38*, pp. 147250–147281, 2025a.

Yehonathan Refael, Jonathan Svirsky, Boris Shustin, Wasim Huleihel, and Ofir Lindenbaum. Adarankgrad: Adaptive gradient rank and moments for memory-efficient LLMs training and fine-tuning. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Thomas Robert, Mher Safaryan, Ionut-Vlad Modoranu, and Dan Alistarh. LDAdam: Adaptive optimization from low-dimensional gradient statistics. In *The Thirteenth International Conference on Learning Representations*, 2025.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: an adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 4463–4473, 2019.

G. W. Stewart. The efficient generation of random orthogonal matrices with an application to condition estimators. *SIAM Journal on Numerical Analysis*, 17(3):403–409, 1980.

Yupeng Su, Ruijie Zhang, Ziyue Liu, Yequan Zhao, and Zheng Zhang. Muonq: Enhancing low-bit muon quantization via directional fidelity optimization. *ArXiv e-prints*, arXiv:2605.11396, 2026.

Zhengbo Wang, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. Taming momentum: Rethinking optimizer states through low-rank approximation. In *The Fourteenth International Conference on Learning Representations*, 2026.

Huaijin Wu, Bingrui Li, Yebin Yang, Yi Tu, Zhanpeng Zhou, Jianfei Chen, and Junchi Yan. Achieving low-bit muon through subspace preservation and grid quantization. In *The Fourteenth International Conference on Learning Representations*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *ArXiv e-prints*, arXiv:2505.09388, 2025a.

Menglin Yang, Jialin Chen, Jinkai Tao, Yifei Zhang, Jiahong Liu, Jiasheng Zhang, Qiyao Ma, Harshit Verma, Regina Zhang, Min Zhou, Irwin King, and Rex Ying. Low-rank adaptation for foundation models: A comprehensive review. *ArXiv e-prints*, arXiv:2501.00365, 2025b.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

Haochen Zhang, Junze Yin, Guanchu Wang, Zirui Liu, Lin Yang, Tianyi Zhang, Anshumali Shrivastava, and Vladimir Braverman. Breaking the frozen subspace: Importance sampling for low-rank optimization in LLM pretraining. In *Advances in Neural Information Processing Systems 38*, pp. 10122–10174, 2025.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. GaLore: Memory-efficient LLM training by gradient low-rank projection. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 61121–61143, 2024.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *ArXiv e-prints*, arXiv:2311.07911, 2023.

Hanqing Zhu, Zhenyu Zhang, Wenyan Cong, Xi Liu, Sem Park, Vikas Chandra, Bo Long, David Z. Pan, Zhangyang Wang, and Jinwon Lee. Apollo: Sgd-like memory, adamw-level performance. In *Proceedings of Machine Learning and Systems*, volume 7. MLSys, 2025.

Algorithm 2 Single-Matrix GaLore-Muon

Input: Initial point $X_0 \in \mathbb{R}^{m\times n}$ with $m \le n$, data distribution $\mathcal{D}$, target horizon T, stepsize η, momentum parameter $\beta \in (0, 1]$, projection rank k, and block length τ, with $S = \lceil T/\tau \rceil$

Initialize $H_{-1} = 0$.
for $s = 0, 1, \dots, S-1$ **do**
  for $t = s\tau, s\tau+1, \dots, \min\{(s+1)\tau, T\} - 1$ **do**
    Sample $\xi_t \sim \mathcal{D}$ and compute $G_t = \nabla f(X_t; \xi_t)$
    if $t = s\tau$ **then**
      Compute $G_t = U_t \Sigma_t V_t^\top$ and set $B_s = U_t[:, 1:k]$
    end if
    Compute $\widehat{G}_t = B_s^\top G_t$
    Update $H_t = (1-\beta)H_{t-1} + \beta \widehat{G}_t$
    Update $X_{t+1} = X_t - \eta B_s \,\mathrm{msign}(H_t)$
  end for
end for
return X_T.

A Additional Experimental Details

A.1 Implementation Details

For randomized basis generation, we use a sign-corrected QR factorization (Mezzadri, 2007). For DRAM with $q_s > 0$, to sample P_s according to (4), we draw a fresh $Z_s \in \mathbb{R}^{m\times q_s}$ with i.i.d. $\mathcal{N}(0, 1)$ entries and project it onto the orthogonal complement of R_s as $\widetilde{Z}_s = Z_s - R_s(R_s^\top Z_s)$. We then compute a thin QR factorization $\widetilde{Z}_s = \widetilde{P}_s T_s$ and apply the standard sign correction to obtain P_s. For $q_s = 0$, we set P_s empty and skip sampling and QR. This produces a Haar-uniform orthonormal q_s-frame in $\mathrm{Range}(I - R_s R_s^\top)$. For GoLore-Muon, we apply the same procedure to an i.i.d. Gaussian matrix of the target rank. For completeness, we present the Muon instantiations of two representative subspace baselines, namely GaLore-Muon in Algorithm 2 and GoLore-Muon in Algorithm 3. At subspace refreshes, GaLore-Muon and GoLore-Muon follow the momentum-state handling conventions of their respective implementations (Zhao et al., 2024; He et al., 2025). GaLore-Muon retains the compact momentum when replacing the principal basis, while GoLore-Muon transports the previous compact state into the new basis through $B_s^\top B_{s-1}$.

All pre-training experiments use four GPUs with PyTorch DistributedDataParallel and NCCL, while all fine-tuning experiments use a single GPU. Experiments were conducted on NVIDIA RTX PRO 6000 Blackwell Server Edition GPUs, running Ubuntu 22.04 with PyTorch and CUDA 13.0.

A.2 Robustness Across Total Ranks and Source Trajectories

We further evaluate the robustness of the stage-wise rank-allocation preference across total ranks and source trajectories. In addition to rank 256 in Fig. 1(b)–(c), we repeat the 200-step interventions along Full-Muon and GaLore-Muon trajectories with total ranks $k = r + q \in \{64, 128\}$. For each rank, we evaluate random-rank fractions $q/k \in \{0, 0.25, 0.5, 0.75, 1\}$ over three independent seeds. All branches use a Muon learning rate of 10^{-2} and a learning rate of 5×10^{-3} for the parameters updated by AdamW. The results are shown in Fig. 3. Across ranks 64, 128, and 256 and both source trajectories, the preference consistently shifts from principal-heavy allocations early in training toward random-heavy allocations later.

A.3 Pre-training Hyperparameters

All models are trained from scratch on C4 with sequence length 1024, global batch size 512, and bfloat16 precision. To improve the reliability of the results, we use scale-specific search grids and report the best-performing configuration for each method. For Muon-based methods, we search the Muon learning rate over $\{0.8, 1.0, 1.2\} \times 10^{-2}$ for LLaMA-60M, $\{0.6, 0.8, 1.0\} \times 10^{-2}$ for LLaMA-130M, and $\{4, 6, 8\} \times 10^{-3}$ for LLaMA-350M. The selected learning rates are summarized in Table 6. For projection-based methods, we refresh the projection every 50 steps for LLaMA-60M

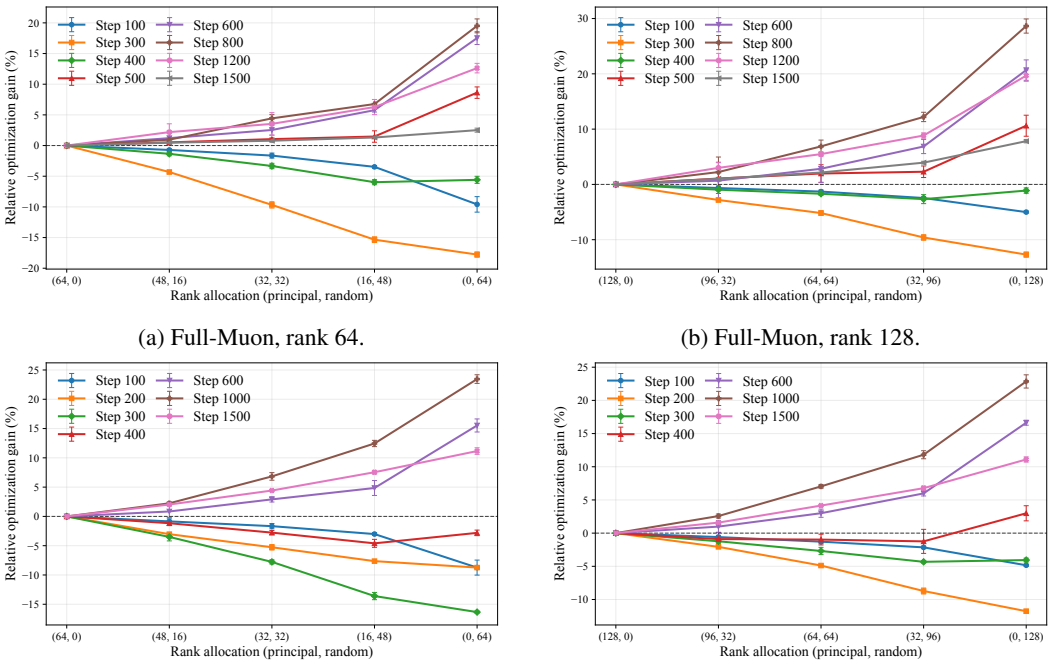
Algorithm 3 Single-Matrix GoLore-Muon

Input: Initial point $X_0 \in \mathbb{R}^{m\times n}$ with $m \le n$, data distribution $\mathcal{D}$, target horizon T, stepsize η, momentum parameter $\beta \in (0, 1]$, projection rank k, and block length τ, with $S = \lceil T/\tau \rceil$

Initialize $H_{-1} = 0$.
for $s = 0, 1, \ldots, S-1$ **do**
  for $t = s\tau, s\tau+1, \ldots, \min\{(s+1)\tau, T\} - 1$ **do**
    Sample $\xi_t \sim \mathcal{D}$ and compute $G_t = \nabla f(X_t; \xi_t)$
    if $t = s\tau$ **then**
      Sample $B_s \sim \mathrm{Unif}\{B \in \mathbb{R}^{m\times k} : B^\top B = I_k\}$
    end if
    Compute $\widehat{G}_t = B_s^\top G_t$
    Update the compact momentum

$$H_t = \begin{cases} (1-\beta) B_s^\top B_{s-1} H_{t-1} + \beta \widehat{G}_t, & t = s\tau,\ s > 0, \\ (1-\beta) H_{t-1} + \beta \widehat{G}_t, & \text{otherwise.} \end{cases}$$

    Update $X_{t+1} = X_t - \eta B_s \,\mathrm{msign}(H_t)$
  end for
end for
return X_T.



(a) Full-Muon, rank 64.

(b) Full-Muon, rank 128.

(c) GaLore-Muon, rank 64.

(d) GaLore-Muon, rank 128.

Figure 3: **Stage-wise rank-allocation preference across total ranks and source trajectories.** Relative gain in 200-step validation-loss reduction over the principal-only allocation across principal-random rank allocations. Results are averaged over three independent seeds.

and every 100 steps for LLaMA-130M and LLaMA-350M, with projection configurations shown in Table 7. For GUM, we fix the projection rank at 128 and choose the full-rank update probability p so that its expected memory cost approximately matches that of the other memory-efficient methods.

Table 6: Learning rates for LLaMA pre-training on C4. For Muon-based methods, entries report the Muon learning rate; AdamW-updated parameters use 5×10^{-3}, 3×10^{-3}, and 2×10^{-3} for LLaMA-60M, 130M, and 350M, respectively.

Model	AdamW	Full-Muon	GaLore-Muon	GoLore-Muon	GUM	DRAM-F	DRAM
LLaMA-60M	5×10^{-3}	1×10^{-2}	1×10^{-2}	1×10^{-2}	1×10^{-2}	1.2×10^{-2}	8×10^{-3}
LLaMA-130M	3×10^{-3}	1×10^{-2}	1×10^{-2}	6×10^{-3}	6×10^{-3}	1×10^{-2}	1×10^{-2}
LLaMA-350M	2×10^{-3}	6×10^{-3}	8×10^{-3}	6×10^{-3}	8×10^{-3}	8×10^{-3}	8×10^{-3}

Table 7: Projection ranks for pre-training experiments. For GUM, p is the full-rank correction probability. For DRAM, (r, q) denotes the principal and random ranks, with DRAM-F fixed at $(128, 128)$ and DRAM linearly reallocating from $(255, 1)$ to $(0, 256)$.

Model	GaLore-Muon	GoLore-Muon	GUM	DRAM-F	DRAM
LLaMA-60M	256	256	128 ($p = 0.53$)	$(128, 128)$	$(255, 1) \to (0, 256)$
LLaMA-130M	256	256	128 ($p = 0.32$)	$(128, 128)$	$(255, 1) \to (0, 256)$
LLaMA-350M	256	256	128 ($p = 0.23$)	$(128, 128)$	$(255, 1) \to (0, 256)$

A.4 Fine-tuning Hyperparameters

For GPT4-LLM-Cleaned, we use the version released by teknium[1]. For each task–backbone pair, we tune η_{Muon} for the weights optimized by Muon and η_{AdamW} for the remaining parameters updated by AdamW. All methods use the same learning-rate grid within each comparison. Table 8 reports the search spaces, and Table 9 reports the selected learning-rate pairs corresponding to Table 4. Projection bases are refreshed every 100 steps for IFEval and every 50 steps for GSM8K and MBPP. For GUM, we use rank 128 with full-rank correction probabilities $p = 0.156$, 0.149, and 0.162 for Qwen3-8B, LLaMA-3.1-8B, and Gemma-2-9B, respectively, chosen to approximately match the expected optimizer-state storage of the rank-512 projection methods.

For instruction following, we fine-tune on 54,546 GPT4-LLM-Cleaned examples with a global batch size of 16 and evaluate on the official 541 IFEval prompts. For mathematical reasoning, we fine-tune on 7,473 GSM8K examples with a global batch size of 16 and evaluate on the official 1,319-example test set. For code generation, we fine-tune on 374 MBPP tasks with a global batch size of 1 and evaluate on the official 500-task test set. All experiments use one epoch, bfloat16 precision, and cosine learning-rate decay.

A.5 GPU Memory Usage

We report the peak GPU memory allocated during full-parameter fine-tuning in Table 10. DRAM reduces peak GPU memory by 17.9–20.7% relative to Full-Muon across the nine fine-tuning settings. Its peak memory differs from that of rank-matched GaLore-Muon and GoLore-Muon by at most 0.04 GiB across the nine settings, indicating negligible additional memory overhead from the principal-random construction and stage-wise rank reallocation. GUM exhibits slightly higher peak memory across all settings, consistent with its probabilistic full-rank corrections.

B Detailed Convergence Analysis

This appendix provides the bound underlying Theorem 1 and Corollary 1, together with their proofs.

Since the target horizon T need not be divisible by the block length τ, the final block of the actual run may contain fewer than τ updates. We therefore set $S = \lceil T/\tau \rceil$ and run Algorithm 1 for exactly T updates. To reuse the complete-block analysis below, we introduce the auxiliary horizon $\widetilde{T} := S\tau$ and consider a conceptual continuation of the final block from T to $\widetilde{T}$. Throughout this appendix, the rank-allocation schedule is prescribed independently of the optimization randomness.

[1] https://huggingface.co/datasets/teknium/GPT4-LLM-Cleaned

Table 8: Learning-rate search spaces for the fine-tuning experiments.

Task	Backbone	η_{Muon}	η_{AdamW}
IFEval	All	$\{1,3,5,8,10,15,20,25,30\}\times10^{-5}$	3×10^{-5}
GSM8K	Qwen3-8B	$\{1,3,5,8,10\}\times10^{-5}$	$\{1,3\}\times10^{-5}$
GSM8K	LLaMA-3.1-8B, Gemma-2-9B	$\{1,3,5,8,10,15,20,25,30\}\times10^{-5}$	$\{1,3\}\times10^{-5}$
MBPP	Qwen3-8B	$\{1,5,10,30,50,80,100\}\times10^{-7}$	$\{1,5,10\}\times10^{-7}$
MBPP	LLaMA-3.1-8B, Gemma-2-9B	$\{1,1.5,2,2.5,3\}\times10^{-4}$	$\{1,3\}\times10^{-5}$

Table 9: Selected learning-rate pairs $(\eta_{\mathrm{Muon}},\eta_{\mathrm{AdamW}})$ for Table 4, in units of 10^{-5}.

Task	Backbone	Full-Muon	GaLore-Muon	GoLore-Muon	GUM	DRAM
IFEval	Qwen3-8B	$(25,3)$	$(25,3)$	$(30,3)$	$(25,3)$	$(30,3)$
	LLaMA-3.1-8B	$(30,3)$	$(30,3)$	$(25,3)$	$(25,3)$	$(25,3)$
	Gemma-2-9B	$(10,3)$	$(15,3)$	$(25,3)$	$(15,3)$	$(15,3)$
GSM8K	Qwen3-8B	$(8,1)$	$(10,1)$	$(10,1)$	$(8,3)$	$(10,1)$
	LLaMA-3.1-8B	$(8,3)$	$(10,1)$	$(10,3)$	$(8,3)$	$(10,3)$
	Gemma-2-9B	$(15,3)$	$(30,1)$	$(30,3)$	$(30,3)$	$(25,1)$
MBPP	Qwen3-8B	$(0.5,0.1)$	$(0.05,0.1)$	$(0.05,0.1)$	$(0.01,0.1)$	$(0.1,0.1)$
	LLaMA-3.1-8B	$(10,3)$	$(10,3)$	$(25,1)$	$(30,1)$	$(30,1)$
	Gemma-2-9B	$(10,1)$	$(15,3)$	$(15,3)$	$(20,1)$	$(15,3)$

For the complete-block extension, define $\mathcal{I}_s := \{s\tau,\dots,(s+1)\tau-1\}$, $k_s = r_s+q_s$, $\Delta_s := \mathbb{E}[F(X_{s\tau})-F(X_{(s+1)\tau})]$ for $s=0,\dots,S-1$, and $\Delta = F(X_0)-F^*$.

For the stochastic setting, let $\delta_s = q_s/(m-r_s)$, where $0\le r_s<m$ and $1\le q_s\le m-r_s$. For the deterministic setting, we allow $0\le r_s\le m$, $0\le q_s\le m-r_s$, $k_s>0$ and define $\gamma_s = k_s/m$.

B.1 Finite-Time Bound and Parameter Choices

Theorem 2 *Let $\widetilde{T} = S\tau$ and $c_\beta := 1-\frac{2(1-\beta)}{\beta\tau}$. Suppose Assumptions 1–4 hold and $c_\beta>0$. For $\sigma>0$, assume $0\le r_s<m$, $1\le q_s\le m-r_s$ for all s, and $\delta_{s+1}\ge\delta_s$ for $s=0,\dots,S-2$. Then the conceptual complete-block extension of Algorithm 1 satisfies*

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \le \frac{\Delta}{\eta c_\beta \widetilde{T}\delta_0} + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta}+\frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\delta_s} + \frac{2\sigma}{c_\beta}\sqrt{\frac{\beta}{2-\beta}}\cdot\frac{1}{S}\sum_{s=0}^{S-1}\frac{\sqrt{k_s}}{\delta_s} + \frac{L_*\eta(\tau-1)}{2S}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s}.$$

For $\sigma=0$, the rank conditions can instead be relaxed to $0\le r_s\le m$, $0\le q_s\le m-r_s$, and $k_s=r_s+q_s>0$ for all s. If $\gamma_{s+1}\ge\gamma_s$ for $s=0,\dots,S-2$, then the conceptual complete-block extension satisfies

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \le \frac{\Delta}{\eta c_\beta \widetilde{T}\gamma_0} + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta}+\frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\gamma_s} + \frac{L_*\eta(\tau-1)}{2S}\sum_{s=0}^{S-1}\frac{1+\gamma_s}{\gamma_s}.$$

The proof combines compact Muon descent, blockwise momentum tracking, and a projected-to-full-gradient conversion.

Table 10: Peak allocated GPU memory (GiB) during full-parameter fine-tuning.

Task	Backbone	Full-Muon	GaLore-Muon	GoLore-Muon	GUM	DRAM
IFEval	Qwen3-8B	57.70	47.38	47.37	47.45	47.38
	LLaMA-3.1-8B	50.86	40.36	40.36	41.31	40.35
	Gemma-2-9B	60.25	48.04	48.08	48.83	48.07
GSM8K	Qwen3-8B	52.86	42.52	42.51	42.83	42.51
	LLaMA-3.1-8B	50.86	40.36	40.35	40.73	40.35
	Gemma-2-9B	61.45	49.24	49.29	49.67	49.28
MBPP	Qwen3-8B	52.80	42.47	42.47	43.23	42.47
	LLaMA-3.1-8B	50.81	40.31	40.31	40.71	40.31
	Gemma-2-9B	60.25	48.03	48.08	48.45	48.06

Parameter choices for Theorem 1. Fix $\beta_0 \in (0, 1]$ independently of T. For $\sigma > 0$, fix an a priori rank cap $1 \le k_{\max} \le m$ and assume

$$T \ge \max\left\{ \left\lceil \frac{3}{\beta_0} \right\rceil, \left\lceil \frac{9k_{\max}\sigma^2}{L_*\Delta} \right\rceil \right\}.$$

Set

$$\beta = \min\left\{ \beta_0, \frac{\sqrt{L_*\Delta}}{\sqrt{k_{\max}}\sigma\sqrt{T}} \right\}, \quad \tau = \left\lceil \frac{3}{\beta} \right\rceil, \quad S = \left\lceil \frac{T}{\tau} \right\rceil, \quad \widetilde{T} = S\tau, \quad \eta = \sqrt{\frac{\Delta\beta}{5L_*\widetilde{T}}}.$$

After S is determined, let $\{(r_s, q_s)\}_{s=0}^{S-1}$ be any prescribed rank-allocation schedule fixed independently of the optimization randomness and satisfying the stochastic conditions of Theorem 1, including $k_s \le k_{\max}$. Thus, the schedule does not affect the choice of β, τ, or S. The horizon condition ensures $\beta \ge 3/T$: if $\beta = \beta_0$, this follows from $T \ge 3/\beta_0$; otherwise, $T \ge 9k_{\max}\sigma^2/(L_*\Delta)$ gives $\sqrt{L_*\Delta}/(\sqrt{k_{\max}}\sigma\sqrt{T}) \ge 3/T$. Hence $\tau \le T$, so $T \le \widetilde{T} < T + \tau \le 2T$. Moreover, $\beta\tau \ge 3$, $\tau - 1 < 3/\beta$, and $c_\beta \ge 1/3$.

For $\sigma = 0$, assume $T \ge \lceil 3/\beta_0 \rceil$ and set

$$\beta = \beta_0, \qquad \tau = \left\lceil \frac{3}{\beta_0} \right\rceil, \qquad S = \left\lceil \frac{T}{\tau} \right\rceil, \qquad \widetilde{T} = S\tau, \qquad \eta = \sqrt{\frac{\Delta\beta_0}{5L_*\widetilde{T}}}.$$

After S is determined, let $\{(r_s, q_s)\}_{s=0}^{S-1}$ be any prescribed rank-allocation schedule satisfying the deterministic conditions of Theorem 1. Here $\tau \le T$, $T \le \widetilde{T} < T + \tau \le 2T$, and $c_\beta \ge 1/3$.

B.2 Supporting Lemmas

Lemma 2 (One-step descent for DRAM) *Suppose Assumptions 3 and 4 hold. Fix a block s and an iteration $t \in \mathcal{I}_s$. Let $B_s = [R_s\ P_s] \in \mathbb{R}^{m\times k_s}$ satisfy $B_s^\top B_s = I_{k_s}$ and let $H_t \in \mathbb{R}^{k_s\times n}$. If $X_{t+1} = X_t - \eta B_s \operatorname{msign}(H_t)$, then*

$$F(X_{t+1}) \le F(X_t) - \eta\|B_s^\top \nabla F(X_t)\|_* + 2\eta\|H_t - B_s^\top \nabla F(X_t)\|_* + \frac{L_*\eta^2}{2}.$$

Proof. Define the lifted compact momentum $\widehat{M}_t := B_s H_t$. Using the compact Muon identity, $\operatorname{msign}(\widehat{M}_t) = B_s \operatorname{msign}(H_t)$, the update can be written as $X_{t+1} = X_t - \eta \operatorname{msign}(\widehat{M}_t)$. By nuclear smoothness in Assumption 3 and $X_{t+1} = X_t - \eta \operatorname{msign}(\widehat{M}_t)$, we obtain

$$F(X_{t+1}) \le F(X_t) - \eta\langle \nabla F(X_t),\ \operatorname{msign}(\widehat{M}_t)\rangle_F + \frac{L_*\eta^2}{2}\|\operatorname{msign}(\widehat{M}_t)\|_{\mathbf{op}}^2.$$

By definition of $\operatorname{msign}(\cdot)$, we have $\|\operatorname{msign}(\widehat{M}_t)\|_{\mathbf{op}} \le 1$. Hence

$$F(X_{t+1}) \le F(X_t) - \eta\langle \nabla F(X_t),\ \operatorname{msign}(\widehat{M}_t)\rangle_F + \frac{L_*\eta^2}{2}.$$

It remains to lower bound the inner product. Using $\operatorname{msign}(\widehat{M}_t) = B_s \operatorname{msign}(H_t)$, it follows that

$$\begin{aligned}\langle \nabla F(X_t), \operatorname{msign}(\widehat{M}_t)\rangle_F &= \langle \nabla F(X_t), B_s \operatorname{msign}(H_t)\rangle_F = \langle B_s^\top \nabla F(X_t), \operatorname{msign}(H_t)\rangle_F \\ &= \langle H_t, \operatorname{msign}(H_t)\rangle_F + \langle B_s^\top \nabla F(X_t) - H_t, \operatorname{msign}(H_t)\rangle_F.\end{aligned}$$

Under Assumption 4, we obtain $\langle H_t, \operatorname{msign}(H_t)\rangle_F = \|H_t\|_*$ and $\|\operatorname{msign}(H_t)\|_{\mathbf{op}} \le 1$. Therefore, by nuclear/operator norm duality, $\langle B_s^\top \nabla F(X_t) - H_t, \operatorname{msign}(H_t)\rangle_F \ge -\|B_s^\top \nabla F(X_t) - H_t\|_*$. Hence

$$\langle \nabla F(X_t), \operatorname{msign}(\widehat{M}_t)\rangle_F \ge \|H_t\|_* - \|B_s^\top \nabla F(X_t) - H_t\|_*.$$

By the triangle inequality $\|H_t\|_* \ge \|B_s^\top \nabla F(X_t)\|_* - \|H_t - B_s^\top \nabla F(X_t)\|_*$, we get

$$\langle \nabla F(X_t), \operatorname{msign}(\widehat{M}_t)\rangle_F \ge \|B_s^\top \nabla F(X_t)\|_* - 2\|H_t - B_s^\top \nabla F(X_t)\|_*.$$

Combining this inequality with the smoothness bound yields

$$F(X_{t+1}) \le F(X_t) - \eta\|B_s^\top \nabla F(X_t)\|_* + 2\eta\|H_t - B_s^\top \nabla F(X_t)\|_* + \frac{L_*\eta^2}{2}.$$

This proves the claim. □

Lemma 3 (Blockwise compact Muon tracking) *Suppose Assumptions 2–4 hold. Fix a block s, and let $B_s = [R_s\ P_s]$ be the blockwise basis used by the conceptual complete-block extension of Algorithm 1. For $j = 0, \dots, \tau - 1$, define*

$$A_j := B_s^\top \nabla F(X_{s\tau+j}), \qquad \widehat{G}_j := B_s^\top G_{s\tau+j}, \qquad H_j := H_{s\tau+j}.$$

Let $\lambda := 1 - \beta$. Then

$$\sum_{j=0}^{\tau-1} \mathbb{E}\left[\|H_j - A_j\|_*\right] \le \frac{1-\beta}{\beta\tau} \sum_{j=0}^{\tau-1} \mathbb{E}\left[\|A_j\|_*\right] + \frac{2(1-\beta)}{\beta}(\tau-1)L_*\eta + \tau\sqrt{k_s}\sigma\sqrt{\frac{\beta}{2-\beta}}.$$

Proof. Within block s, the basis B_s is fixed after the block start. By Algorithm 1,

$$H_0 = \beta\widehat{G}_0, \qquad H_j = (1-\beta)H_{j-1} + \beta\widehat{G}_j, \quad j \ge 1.$$

Let $E_j := H_j - A_j$ and $\zeta_j := \widehat{G}_j - A_j = B_s^\top (G_{s\tau+j} - \nabla F(X_{s\tau+j}))$. Then $E_0 = -\lambda A_0 + \beta\zeta_0$, and for $j \ge 1$,

$$\begin{aligned}E_j &= (1-\beta)H_{j-1} + \beta\widehat{G}_j - A_j \\ &= \lambda E_{j-1} + \lambda(A_{j-1} - A_j) + \beta\zeta_j.\end{aligned}$$

Unrolling this recursion gives

$$E_j = -\lambda^{j+1}A_0 + \sum_{i=1}^{j} \lambda^{j-i+1}(A_{i-1} - A_i) + \beta\sum_{i=0}^{j} \lambda^{j-i}\zeta_i.$$

Taking nuclear norms, using the triangle inequality and summing over $j = 0, \cdots, \tau - 1$, we obtain

$$\sum_{j=0}^{\tau-1} \|E_j\|_* \le \sum_{j=0}^{\tau-1} \lambda^{j+1}\|A_0\|_* + \sum_{j=0}^{\tau-1}\sum_{i=1}^{j} \lambda^{j-i+1}\|A_{i-1} - A_i\|_* + \sum_{j=0}^{\tau-1} \left\| \beta \sum_{i=0}^{j} \lambda^{j-i}\zeta_i \right\|_*.$$

We control the three terms separately.

For the initial term, summing the geometric series yields

$$\sum_{j=0}^{\tau-1} \lambda^{j+1}\|A_0\|_* \le \frac{\lambda}{\beta}\|A_0\|_*.$$

To relate $\|A_0\|_*$ to the average projected gradient over the block, we use telescoping: for every j,

$$\|A_0\|_* \le \|A_j\|_* + \sum_{i=1}^{j} \|A_{i-1} - A_i\|_*.$$

Averaging this inequality over $j = 0, \dots, \tau - 1$ gives

$$\|A_0\|_* \le \frac{1}{\tau} \sum_{j=0}^{\tau-1} \|A_j\|_* + \sum_{i=1}^{\tau-1} \|A_i - A_{i-1}\|_*.$$

Next, we control the variation of A_j within the block. By the definition of A_j, we obtain

$$\begin{aligned} \|A_i - A_{i-1}\|_* &= \|B_s^\top (\nabla F(X_{s\tau+i}) - \nabla F(X_{s\tau+i-1}))\|_* \\ &\le \|\nabla F(X_{s\tau+i}) - \nabla F(X_{s\tau+i-1})\|_* \\ &\le L_* \|X_{s\tau+i} - X_{s\tau+i-1}\|_{\mathbf{op}} \le L_* \eta. \end{aligned}$$

Substituting this drift bound into the previous estimate gives

$$\sum_{j=0}^{\tau-1} \lambda^{j+1} \|A_0\|_* \le \frac{\lambda}{\beta\tau} \sum_{j=0}^{\tau-1} \|A_j\|_* + \frac{\lambda}{\beta}(\tau - 1) L_* \eta.$$

Second, we bound the accumulated drift term. Using again $\|A_i - A_{i-1}\|_* \le L_* \eta$, we obtain

$$\sum_{j=0}^{\tau-1} \sum_{i=1}^{j} \lambda^{j-i+1} \|A_{i-1} - A_i\|_* \le \sum_{j=1}^{\tau-1} \sum_{i=1}^{j} \lambda^{j-i+1} L_* \eta \le \sum_{j=1}^{\tau-1} \frac{\lambda}{\beta} L_* \eta \le \frac{\lambda}{\beta}(\tau - 1) L_* \eta.$$

Third, we control the stochastic noise term. Since the matrix inside the nuclear norm lies in $\mathbb{R}^{k_s \times n}$, its rank is at most k_s. Hence

$$\mathbb{E} \left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_* \le \sqrt{k_s}\, \mathbb{E} \left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_F \le \sqrt{k_s} \left(\mathbb{E} \left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_F^2 \right)^{1/2}.$$

We now bound the Frobenius second moment.

For $0 \le i < f \le j$, note that $f \ge 1$. By iteration $s\tau + f$, the block basis B_s has already been constructed, and both B_s and ζ_i are $\mathcal{F}_{s\tau+f}$-measurable. Moreover, by Assumption 2,

$$\mathbb{E}[\zeta_f \mid \mathcal{F}_{s\tau+f}] = B_s^\top \mathbb{E}[G_{s\tau+f} - \nabla F(X_{s\tau+f}) \mid \mathcal{F}_{s\tau+f}] = 0.$$

Hence, by the tower property, we obtain $\mathbb{E}\langle \zeta_i, \zeta_f \rangle_F = \mathbb{E}\big[\langle \zeta_i, \mathbb{E}[\zeta_f \mid \mathcal{F}_{s\tau+f}] \rangle_F\big] = 0$. Therefore,

$$\mathbb{E} \left[\left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_F^2 \right] = \beta^2 \sum_{i=0}^{j} \lambda^{2(j-i)} \mathbb{E} \|\zeta_i\|_F^2.$$

It remains to bound the individual second moments. Although B_s depends on the block-start stochastic gradient, no independence is needed here. Since $B_s^\top B_s = I$, the projection is contractive in Frobenius norm, and hence

$$\|\zeta_i\|_F = \big\| B_s^\top \big(G_{s\tau+i} - \nabla F(X_{s\tau+i}) \big) \big\|_F \le \|G_{s\tau+i} - \nabla F(X_{s\tau+i})\|_F.$$

Taking expectations and applying Assumption 2 gives $\mathbb{E}\|\zeta_i\|_F^2 \le \sigma^2$. Consequently,

$$\mathbb{E} \left[\left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_F^2 \right] \le \beta^2 \sigma^2 \sum_{i=0}^{j} \lambda^{2(j-i)} \le \frac{\beta}{2 - \beta} \sigma^2,$$

where we used $1 - \lambda^2 = 1 - (1 - \beta)^2 = \beta(2 - \beta)$. Finally, since the matrix inside the norm has at most k_s rows,

$$\mathbb{E} \left[\left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_* \right] \le \sqrt{k_s} \left(\mathbb{E} \left[\left\| \beta \sum_{i=0}^{j} \lambda^{j-i} \zeta_i \right\|_F^2 \right] \right)^{1/2} \le \sqrt{k_s} \sigma \sqrt{\frac{\beta}{2 - \beta}}.$$

Combining the bounds for the initial-error term, drift, and noise terms, we get

$$\sum_{j=0}^{\tau-1} \mathbb{E} \left[\|E_j\|_* \right] \le \frac{\lambda}{\beta\tau} \sum_{j=0}^{\tau-1} \mathbb{E} \left[\|A_j\|_* \right] + \frac{2\lambda}{\beta}(\tau - 1) L_* \eta + \tau \sqrt{k_s} \sigma \sqrt{\frac{\beta}{2 - \beta}}.$$

Finally, substituting $\lambda = 1 - \beta$ proves the claim. □

Lemma 4 (Weighted projected stationarity) *Suppose Assumptions 1–4 hold. Assume $0 \le r_s < m$ and $1 \le q_s \le m - r_s$ for all s, and define $\delta_s = q_s/(m - r_s)$. Let $\widetilde{T} = S\tau$, $\Delta = F(X_0) - F^*$, and $c_\beta = 1 - \frac{2(1-\beta)}{\beta\tau}$. If $c_\beta > 0$, then the iterates in the conceptual complete-block extension of Algorithm 1 satisfy*

$$\begin{aligned}&\frac{1}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\delta_s}\sum_{j=0}^{\tau-1}\mathbb{E}\left[\|B_s^\top \nabla F(X_{s\tau+j})\|_*\right]\\ \le&\frac{1}{\eta c_\beta \widetilde{T}}\sum_{s=0}^{S-1}\frac{\Delta_s}{\delta_s} + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta} + \frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\delta_s} + \frac{2\sigma}{c_\beta}\sqrt{\frac{\beta}{2-\beta}}\cdot\frac{1}{S}\sum_{s=0}^{S-1}\frac{\sqrt{k_s}}{\delta_s}.\end{aligned} \tag{7}$$

Proof. Fix a block s. To simplify notation, define the accumulated projected gradient norm in this block as $S_s := \sum_{j=0}^{\tau-1}\mathbb{E}\left[\|B_s^\top \nabla F(X_{s\tau+j})\|_*\right]$, and define the blockwise function decrease as $\Delta_s = \mathbb{E}\left[F(X_{s\tau}) - F(X_{(s+1)\tau})\right]$.

We first apply the one-step descent inequality in Lemma 2 to all iterations in block s. Summing over $j = 0, \dots, \tau - 1$ and taking expectation gives

$$\eta S_s \le \Delta_s + 2\eta\sum_{j=0}^{\tau-1}\mathbb{E}\left[\|H_{s\tau+j} - B_s^\top \nabla F(X_{s\tau+j})\|_*\right] + \frac{\tau L_*\eta^2}{2}.$$

Using Lemma 3, we obtain

$$\eta S_s \le \Delta_s + 2\eta\left[\frac{1-\beta}{\beta\tau}S_s + \frac{2(1-\beta)}{\beta}(\tau-1)L_*\eta + \tau\sqrt{k_s}\sigma\sqrt{\frac{\beta}{2-\beta}}\right] + \frac{\tau L_*\eta^2}{2}.$$

Moving the S_s term to the left and dividing by ηc_β gives the per-block estimate

$$S_s \le \frac{\Delta_s}{\eta c_\beta} + \frac{4(1-\beta)(\tau-1)L_*\eta}{\beta c_\beta} + \frac{2\tau\sqrt{k_s}\sigma}{c_\beta}\sqrt{\frac{\beta}{2-\beta}} + \frac{\tau L_*\eta}{2c_\beta}, \tag{8}$$

where $c_\beta = 1 - \frac{2(1-\beta)}{\beta\tau}$. Therefore,

$$\begin{aligned}\frac{1}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\delta_s}S_s \le&\frac{1}{\eta c_\beta\widetilde{T}}\sum_{s=0}^{S-1}\frac{\Delta_s}{\delta_s} + \frac{4(1-\beta)(\tau-1)L_*\eta}{\beta c_\beta\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\delta_s}\\ &+ \frac{2\tau\sigma}{c_\beta\widetilde{T}}\sqrt{\frac{\beta}{2-\beta}}\sum_{s=0}^{S-1}\frac{\sqrt{k_s}}{\delta_s} + \frac{\tau L_*\eta}{2c_\beta\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\delta_s}.\end{aligned}$$

Using $\widetilde{T} = S\tau$ gives (7), which completes the proof. □

Lemma 5 (Coverage for projected nuclear stationarity) *Fix a block $s \in \{0, \dots, S-1\}$. Let $\Pi_s := R_s R_s^\top$, $Q_s := I - \Pi_s$ and $B_s = [R_s\ P_s]$, where $R_s^\top R_s = I_{r_s}$ and P_s is sampled as in Algorithm 1. Suppose Assumptions 3 and 4 hold and the iterates belong to the conceptual complete-block extension of Algorithm 1.*

If $1 \le q_s \le m - r_s$ and $\delta_s = q_s/(m - r_s)$, then for any $t = s\tau + j$ with $0 \le j \le \tau - 1$,

$$\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \le \frac{1}{\delta_s}\mathbb{E}\left[\|B_s^\top \nabla F(X_t)\|_*\right] + \frac{1+\delta_s}{\delta_s}L_* j\eta. \tag{9}$$

Moreover, suppose Assumption 2 holds with $\sigma = 0$. If $0 \le q_s \le m - r_s$ with $k_s = r_s + q_s \ge 1$, and $\gamma_s = k_s/m$, then

$$\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \le \frac{1}{\gamma_s}\mathbb{E}\left[\|B_s^\top \nabla F(X_t)\|_*\right] + \frac{1+\gamma_s}{\gamma_s}L_* j\eta. \tag{10}$$

Importantly, the second bound remains valid when $q_s = 0$.

Proof. We first prove the stochastic coverage bound, where $q_s > 0$. Let $A := \nabla F(X_{s\tau})$. Conditional on $(R_s, X_{s\tau})$, the matrix A is fixed and the only remaining randomness is P_s. Since $B_s^\top B_s = I$, the matrices $B_s^\top A$ and $B_s B_s^\top A$ have the same nonzero singular values, hence $\|B_s^\top A\|_* = \|B_s B_s^\top A\|_*$. By Lemma 1,

$$\mathbb{E}_{P_s}[B_s B_s^\top \mid R_s, X_{s\tau}] = \Pi_s + \delta_s Q_s.$$

Therefore, by Jensen's inequality,

$$\begin{aligned}\mathbb{E}_{P_s}\big[\|B_s^\top A\|_* \mid R_s, X_{s\tau}\big] &= \mathbb{E}_{P_s}\big[\|B_s B_s^\top A\|_* \mid R_s, X_{s\tau}\big] \\ &\geq \big\|\mathbb{E}_{P_s}[B_s B_s^\top \mid R_s, X_{s\tau}]A\big\|_* \\ &= \|(\Pi_s + \delta_s Q_s)A\|_* \\ &\geq \delta_s \|A\|_*.\end{aligned}$$

The operator $\Pi_s + \delta_s Q_s$ has singular value 1 on $\mathrm{Range}(\Pi_s)$ and singular value δ_s on $\mathrm{Range}(Q_s)$, so $\sigma_{\min}(\Pi_s + \delta_s Q_s) = \delta_s$. Therefore $\|(\Pi_s + \delta_s Q_s)A\|_* \geq \delta_s \|A\|_*$. Taking expectation over $(R_s, X_{s\tau})$ yields $\mathbb{E}\|B_s^\top \nabla F(X_{s\tau})\|_* \geq \delta_s \mathbb{E}\|\nabla F(X_{s\tau})\|_*$.

We next extend the bound from the block start to any $t = s\tau + j$, where $0 \leq j \leq \tau - 1$. By the triangle inequality and $\|B_s^\top Z\|_* \leq \|Z\|_*$,

$$\begin{aligned}\|B_s^\top \nabla F(X_t)\|_* &\geq \|B_s^\top \nabla F(X_{s\tau})\|_* - \|B_s^\top(\nabla F(X_t) - \nabla F(X_{s\tau}))\|_* \\ &\geq \|B_s^\top \nabla F(X_{s\tau})\|_* - \|\nabla F(X_t) - \nabla F(X_{s\tau})\|_*.\end{aligned}$$

Taking expectation and using the block-start bound gives

$$\mathbb{E}\|B_s^\top \nabla F(X_t)\|_* \geq \delta_s \mathbb{E}\|\nabla F(X_{s\tau})\|_* - \mathbb{E}\|\nabla F(X_t) - \nabla F(X_{s\tau})\|_*.$$

Moreover, $\|\nabla F(X_{s\tau})\|_* \geq \|\nabla F(X_t)\|_* - \|\nabla F(X_t) - \nabla F(X_{s\tau})\|_*$. Combining the two inequalities gives

$$\mathbb{E}\|B_s^\top \nabla F(X_t)\|_* \geq \delta_s \mathbb{E}\|\nabla F(X_t)\|_* - (1 + \delta_s)\mathbb{E}\|\nabla F(X_t) - \nabla F(X_{s\tau})\|_*.$$

By nuclear-smoothness, $\|\nabla F(X_t) - \nabla F(X_{s\tau})\|_* \leq L_* \|X_t - X_{s\tau}\|_{\mathbf{op}}$. Since $B_s^\top B_s = I$ and $\|\operatorname{msign}(H_j)\|_{\mathbf{op}} \leq 1$, for $t = s\tau + j$ we have

$$\|X_t - X_{s\tau}\|_{\mathbf{op}} \leq \sum_{f=s\tau}^{t-1} \|X_{f+1} - X_f\|_{\mathbf{op}} = \eta \sum_{f=s\tau}^{t-1} \|B_s \operatorname{msign}(H_f)\|_{\mathbf{op}} \leq j\eta.$$

Therefore $\mathbb{E}\|\nabla F(X_t) - \nabla F(X_{s\tau})\|_* \leq L_* j\eta$. Substituting this estimate into the previous lower bound gives

$$\mathbb{E}\|B_s^\top \nabla F(X_t)\|_* \geq \delta_s \mathbb{E}\|\nabla F(X_t)\|_* - (1 + \delta_s) L_* j\eta.$$

Rearranging terms proves the stochastic bound (9).

For the deterministic case, Assumption 2 with $\sigma = 0$ implies $G_{s\tau} = \nabla F(X_{s\tau})$ almost surely. Let $A = \nabla F(X_{s\tau})$.

If $r_s = m$, then necessarily $q_s = 0$ and $B_s = R_s$ is an $m \times m$ orthogonal matrix. Hence

$$\|B_s^\top A\|_* = \|A\|_* = \gamma_s \|A\|_*, \qquad \gamma_s = \frac{k_s}{m} = 1.$$

Thus, the desired block-start coverage bound holds directly.

It remains to consider $r_s < m$. Let $A = \nabla F(X_{s\tau}) = U\Sigma V^\top$ with $\sigma_1(A) \geq \cdots \geq \sigma_m(A) \geq 0$. Since R_s is formed by the leading r_s left singular vectors of $G_{s\tau} = A$, we have

$$\|\Pi_s A\|_* = \sum_{i=1}^{r_s} \sigma_i(A).$$

Let $\bar{\delta}_s := q_s/(m - r_s)$. When $q_s = 0$, $B_s B_s^\top = \Pi_s$. When $q_s > 0$, Lemma 1 gives $\mathbb{E}_{P_s}[B_s B_s^\top \mid R_s, X_{s\tau}] = \Pi_s + \bar{\delta}_s Q_s$. Thus, in both cases,

$$\mathbb{E}_{P_s}\big[\|B_s^\top A\|_* \mid R_s, X_{s\tau}\big] \geq \|(\Pi_s + \bar{\delta}_s Q_s)A\|_* = \sum_{i=1}^{r_s} \sigma_i(A) + \bar{\delta}_s \sum_{i=r_s+1}^{m} \sigma_i(A).$$

Since the singular values are nonincreasing, we obtain $\sum_{i=1}^{r_s}\sigma_i(A) \geq \frac{r_s}{m}\|A\|_*$. Therefore,

$$\mathbb{E}_{P_s}\left[\|B_s^\top A\|_* \mid R_s, X_{s\tau}\right] \geq \left[\bar{\delta}_s + (1-\bar{\delta}_s)\frac{r_s}{m}\right]\|A\|_* = \frac{r_s+q_s}{m}\|A\|_* = \gamma_s\|A\|_*.$$

Taking expectation yields $\mathbb{E}\|B_s^\top \nabla F(X_{s\tau})\|_* \geq \gamma_s \mathbb{E}\|\nabla F(X_{s\tau})\|_*$. Repeating the same within-block drift argument used above, with γ_s in place of δ_s, gives

$$\mathbb{E}\|B_s^\top \nabla F(X_t)\|_* \geq \gamma_s \mathbb{E}\|\nabla F(X_t)\|_* - (1+\gamma_s)L_* j\eta,$$

which is equivalent to (10). □

B.3 Proof of Theorem 2

We use Lemma 5 to convert the projected stationarity bound into a full-gradient stationarity bound. Fix a block s and let $t = s\tau + j$, where $0 \leq j \leq \tau - 1$. By Lemma 5,

$$\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \leq \frac{1}{\delta_s}\mathbb{E}\left[\|B_s^\top \nabla F(X_t)\|_*\right] + \frac{1+\delta_s}{\delta_s}L_* j\eta.$$

Averaging this inequality over all iterations of the complete-block extension gives

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \leq \frac{1}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\delta_s}\sum_{j=0}^{\tau-1}\mathbb{E}\left[\|B_s^\top \nabla F(X_{s\tau+j})\|_*\right] + \frac{L_*\eta}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s}\sum_{j=0}^{\tau-1} j. \quad (11)$$

Since $\sum_{j=0}^{\tau-1} j = \frac{\tau(\tau-1)}{2}$ and $\widetilde{T} = S\tau$, the second term becomes

$$\frac{L_*\eta}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s}\cdot\frac{\tau(\tau-1)}{2} = \frac{L_*\eta(\tau-1)}{2S}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s}. \quad (12)$$

Combining (7), (11) and (12) yields

$$\begin{aligned}\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\left[\|\nabla F(X_t)\|_*\right] \leq & \frac{1}{\eta c_\beta \widetilde{T}}\sum_{s=0}^{S-1}\frac{\Delta_s}{\delta_s} + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta} + \frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\delta_s} \\ & + \frac{2\sigma}{c_\beta}\sqrt{\frac{\beta}{2-\beta}}\cdot\frac{1}{S}\sum_{s=0}^{S-1}\frac{\sqrt{k_s}}{\delta_s} + \frac{L_*\eta(\tau-1)}{2S}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s}.\end{aligned} \quad (13)$$

It remains to control the weighted decrease term. Let $F_s := \mathbb{E}[F(X_{s\tau}) - F^*] \geq 0$. Then $\Delta_s = F_s - F_{s+1}$. Since the complementary coverage factor is assumed to be nondecreasing, i.e., $\delta_{s+1} \geq \delta_s$, the sequence $1/\delta_s$ is nonincreasing. Hence, by summation by parts,

$$\sum_{s=0}^{S-1}\frac{\Delta_s}{\delta_s} = \sum_{s=0}^{S-1}\frac{F_s - F_{s+1}}{\delta_s} = \frac{F_0}{\delta_0} - \frac{F_S}{\delta_{S-1}} + \sum_{s=1}^{S-1}\left(\frac{1}{\delta_s} - \frac{1}{\delta_{s-1}}\right)F_s \leq \frac{F_0}{\delta_0} = \frac{\Delta}{\delta_0}. \quad (14)$$

Substituting (14) into (13) completes the proof of the stage-dependent full-gradient stationarity bound.

We then consider the deterministic case. Applying Lemmas 2 and 3 blockwise and setting $\sigma = 0$ gives the same per-block estimate as (8),

$$\sum_{j=0}^{\tau-1}\mathbb{E}\|B_s^\top \nabla F(X_{s\tau+j})\|_* \leq \frac{\Delta_s}{\eta c_\beta} + \frac{4(1-\beta)(\tau-1)L_*\eta}{\beta c_\beta} + \frac{\tau L_*\eta}{2c_\beta}.$$

Multiplying the above inequality by $1/\gamma_s$, summing over $s = 0, \ldots, S-1$, and using $\widetilde{T} = S\tau$, we obtain

$$\begin{aligned}\frac{1}{\widetilde{T}}\sum_{s=0}^{S-1}\frac{1}{\gamma_s}\sum_{j=0}^{\tau-1}\mathbb{E}\|B_s^\top \nabla F(X_{s\tau+j})\|_* \leq & \frac{1}{\eta c_\beta \widetilde{T}}\sum_{s=0}^{S-1}\frac{\Delta_s}{\gamma_s} \\ & + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta} + \frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\gamma_s}.\end{aligned}$$

Recall that $F_s = \mathbb{E}[F(X_{s\tau}) - F^*] \geq 0$ and $\Delta_s = F_s - F_{s+1}$. Since $\gamma_{s+1} \geq \gamma_s$, the sequence $\{1/\gamma_s\}_{s=0}^{S-1}$ is nonincreasing. Hence, following (14), we obtain $\sum_{s=0}^{S-1} \frac{\Delta_s}{\gamma_s} \leq \frac{\Delta}{\gamma_0}$. By (10) in Lemma 5, for $t = s\tau + j$,

$$\mathbb{E}\|\nabla F(X_t)\|_* \leq \frac{1}{\gamma_s}\mathbb{E}\|B_s^\top \nabla F(X_t)\|_* + \frac{1+\gamma_s}{\gamma_s} L_* j\eta.$$

Averaging over all iterations and using $\sum_{j=0}^{\tau-1} j = \frac{\tau(\tau-1)}{2}$ gives

$$\begin{aligned}\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* \leq & \frac{\Delta}{\eta c_\beta \widetilde{T}\gamma_0} + \left[\frac{4(1-\beta)(\tau-1)L_*\eta}{\beta\tau c_\beta} + \frac{L_*\eta}{2c_\beta}\right]\frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\gamma_s} \\ & + \frac{L_*\eta(\tau-1)}{2S}\sum_{s=0}^{S-1}\frac{1+\gamma_s}{\gamma_s},\end{aligned} \tag{15}$$

which proves the deterministic statement.

B.4 Proof of Theorem 1

The actual trajectory and its conceptual complete-block extension coincide through X_T. We first bound the extension and then transfer the result to the actual prefix.

Let $A_\delta := \frac{1}{S}\sum_{s=0}^{S-1}\delta_s^{-1}$, so $\Gamma_\delta = \delta_0^{-1} + A_\delta$. Since $0 < \delta_s \leq 1$,

$$\frac{1}{S}\sum_{s=0}^{S-1}\frac{1+\delta_s}{\delta_s} = 1 + A_\delta \leq \Gamma_\delta.$$

Moreover, since $k_s \leq k_{\max}$,

$$\frac{1}{S}\sum_{s=0}^{S-1}\frac{\sqrt{k_s}}{\delta_s} \leq \sqrt{k_{\max}}A_\delta \leq \sqrt{k_{\max}}\Gamma_\delta.$$

The stochastic parameter choice gives $\tau = \lceil 3/\beta \rceil$, so $c_\beta \geq 1/3$, $\tau - 1 < 3/\beta$, and $(\tau-1)/\tau \leq 1$. Applying Theorem 2, using $A_\delta \leq \Gamma_\delta$, $\delta_0^{-1} \leq \Gamma_\delta$, $\beta \leq 1$, and the bound above gives

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* \leq 3\Gamma_\delta\frac{\Delta}{\eta\widetilde{T}} + 15\Gamma_\delta\frac{L_*\eta}{\beta} + 6\Gamma_\delta\sqrt{k_{\max}}\sigma\sqrt{\beta}.$$

Substituting $\eta = \sqrt{\Delta\beta/(5L_*\widetilde{T})}$ yields

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* \leq 6\Gamma_\delta\sqrt{\frac{5L_*\Delta}{\widetilde{T}\beta}} + 6\Gamma_\delta\sqrt{k_{\max}}\sigma\sqrt{\beta}.$$

Let $a := \sqrt{L_*\Delta/T}$ and $b := \sqrt{k_{\max}}\sigma$. Since $\widetilde{T} \geq T$, the preceding bound gives

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* \leq 6\Gamma_\delta\left(\sqrt{5}\frac{a}{\sqrt{\beta}} + b\sqrt{\beta}\right).$$

By construction, $\beta = \min\{\beta_0, a/b\}$. If $\beta = \beta_0$, then $b\beta_0 \leq a$, and hence $b\sqrt{\beta_0} \leq a/\sqrt{\beta_0}$. If $\beta = a/b$, then $a/\sqrt{\beta} = b\sqrt{\beta} = \sqrt{ab}$. Therefore,

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* \leq 6(\sqrt{5}+1)\Gamma_\delta\left(\frac{a}{\sqrt{\beta_0}} + \sqrt{ab}\right).$$

Since β_0 is fixed and $\sqrt{ab} = (k_{\max}L_*\Delta\sigma^2/T)^{1/4}$,

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\Gamma_\delta\left[\sqrt{\frac{L_*\Delta}{T}} + \left(\frac{k_{\max}L_*\Delta\sigma^2}{T}\right)^{1/4}\right]\right).$$

Since the stationarity terms are nonnegative,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(X_t)\|_* \le \frac{\widetilde{T}}{T}\left[\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_*\right].$$

Using $\widetilde{T} < 2T$ proves the stochastic statement of Theorem 1.

Deterministic case. Let $A_\gamma := \frac{1}{S}\sum_{s=0}^{S-1}\frac{1}{\gamma_s}$, so $\Gamma_\gamma = \gamma_0^{-1} + A_\gamma$. Since $\beta = \beta_0$ is fixed, $\tau = \lceil 3/\beta_0 \rceil = O(1)$ and $c_\beta \ge 1/3$. The deterministic part of Theorem 2 therefore gives

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\Gamma_\gamma\left[\frac{\Delta}{\eta\widetilde{T}} + L_*\eta\right]\right).$$

Substituting $\eta = \sqrt{\Delta\beta_0/(5L_*\widetilde{T})}$ yields

$$\frac{1}{\widetilde{T}}\sum_{t=0}^{\widetilde{T}-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\Gamma_\gamma\sqrt{\frac{L_*\Delta}{\widetilde{T}}}\right).$$

Applying the same prefix inequality and $T \le \widetilde{T} < 2T$ proves

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\Gamma_\gamma\sqrt{\frac{L_*\Delta}{T}}\right).$$

B.5 Proof of Lemma 1

Let $V \in \mathbb{R}^{m\times(m-r)}$ be an orthonormal basis of $\mathrm{Range}(Q)$, so that $Q = VV^\top$. Since P is uniformly sampled from the q-dimensional orthonormal frames in $\mathrm{Range}(Q)$, we can write

$$P = VU, \qquad U \sim \mathrm{Unif}(\mathrm{St}(m-r, q)).$$

By rotational invariance of the Stiefel distribution, $\mathbb{E}[UU^\top]$ is invariant under orthogonal conjugation and hence equals aI_{m-r} for some scalar a. Taking traces gives

$$q = \mathbb{E}[\mathrm{tr}(U^\top U)] = a(m-r),$$

so $a = q/(m-r) = \delta$. Therefore,

$$\mathbb{E}[PP^\top \mid R] = V\mathbb{E}[UU^\top]V^\top = \delta Q.$$

Since $B = [R\ P]$ and $R^\top P = 0$, we obtain $BB^\top = \Pi + PP^\top$, which yields

$$\mathbb{E}[BB^\top \mid R] = \Pi + \delta Q.$$

Next, since $B^\top B = I$, left multiplication by B preserves the singular values of $B^\top A$, and hence $\|B^\top A\|_* = \|BB^\top A\|_*$. By convexity of the nuclear norm and Jensen's inequality,

$$\mathbb{E}\big[\|B^\top A\|_* \mid R\big] = \mathbb{E}\big[\|BB^\top A\|_* \mid R\big] \ge \big\|\mathbb{E}[BB^\top \mid R]A\big\|_* = \|(\Pi + \delta Q)A\|_*.$$

Finally, $\Pi + \delta Q = \delta I + (1-\delta)\Pi$ has smallest singular value δ. Since $\delta > 0$,

$$\|A\|_* = \|(\Pi+\delta Q)^{-1}(\Pi+\delta Q)A\|_* \le \|(\Pi+\delta Q)^{-1}\|_{\mathrm{op}}\|(\Pi+\delta Q)A\|_* = \delta^{-1}\|(\Pi+\delta Q)A\|_*.$$

Therefore, $\|(\Pi+\delta Q)A\|_* \ge \delta\|A\|_*$, which completes the proof.

B.6 Proof of Corollary 1

For fixed ranks $r_s \equiv r$ and $q_s \equiv q$, we have $k_s \equiv k = r + q$. For $\sigma > 0$, choose $k_{\max} = k$. Then $\delta_s \equiv \delta = q/(m-r)$ and hence $\Gamma_\delta = 2/\delta$. Direct substitution into the stochastic part of Theorem 1 gives

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\frac{1}{\delta}\left[\sqrt{\frac{L_*\Delta}{T}} + \left(\frac{L_*\Delta k\sigma^2}{T}\right)^{1/4}\right]\right).$$

For $\sigma = 0$, $\gamma_s \equiv k/m$, so $\Gamma_\gamma = 2m/k$. Substitution into the deterministic part of Theorem 1 gives

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(X_t)\|_* = O\left(\frac{m}{k}\sqrt{\frac{L_*\Delta}{T}}\right),$$

which proves the claim.