ORTHVMS: ORTHOGONAL VARIATIONAL MULTISCALE CORRECTION FOR FULL-RANK UPDATES WITH LOW-RANK OPTIMIZER STATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Training large language models (LLMs) incurs substantial memory overhead, motivating methods that compress optimizer states via low-rank gradient projection. However, projection-based optimizers restrict each update to a compact subspace, discarding the orthogonal gradient component, while maintaining full-size moment states for this residual remains memory-prohibitive. To harness this orthogonal information without sacrificing efficiency, we propose **OrthVMS**, an **Orth**ogonal **V**ariational **M**ulti**S**cale correction framework. Drawing inspiration from variational multiscale modeling (VMS) in computational mechanics, OrthVMS treats the projected space and its orthogonal complement as coarse and fine spaces, respectively. It constructs a full-space preconditioner from two decoupled components: the coarse component maintains compact adaptive moments within the projected space, while the fine component preconditions the orthogonal residual using its historical energy statistics. This residual-driven correction requires only a single persistent vector of additional states. Extensive experiments on LLM benchmarks demonstrate that OrthVMS consistently outperforms existing memory-efficient training methods. Our code is available here: `https://anonymous.4open.science/r/Orthogonal-VMS-C26C`.

1 INTRODUCTION

Large language models (LLMs) have achieved remarkable progress in natural language understanding, text generation, and complex reasoning (Brown et al., 2020, Wei et al., 2022). Training them at scale, however, consumes enormous computational and memory resources, which makes optimizer design central to both training efficiency and stability. AdamW (Loshchilov & Hutter, 2017) has become the prevailing optimizer for large-scale training owing to its adaptive parameter updates and strong empirical performance. This comes at a cost, since AdamW maintains first- and second-moment states for every parameter, incurring heavy memory overhead. In mixed-precision training with FP32 master weights and FP32 moment states, a 1.5B-parameter model needs about 24 GB for its parameters, gradients, and optimizer states alone, and the optimizer states account for about 18 GB when the master weight copy is counted among them (Rajbhandari et al., 2020).

To reduce optimizer memory, Adam-mini (Zhang et al., 2025) shares a second-moment scale within each parameter block instead of estimating element-wise learning rates, but its partition must be tailored to attention heads and module structure. Another line of work compresses weight updates. LoRA (Hu et al., 2021) learns low-rank updates on frozen pretrained weights, saving fine-tuning memory at the cost of update expressiveness. ReLoRA (Lialin et al., 2024) reaches higher-rank updates via periodic merge-and-restart, yet its full-rank warm-up keeps complete optimizer states and thus high peak memory for pretraining from scratch.

Table 1: Comparison of update schemes. Precond. abbreviates preconditioning.

	Coarse space		Fine space	
Method	Update	Precond.	Update	Precond.
Projected update	✓	✓	✗	✗
Gradient supplement	✓	✓	✓	✗
OrthVMS (Ours)	✓	✓	✓	✓

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.

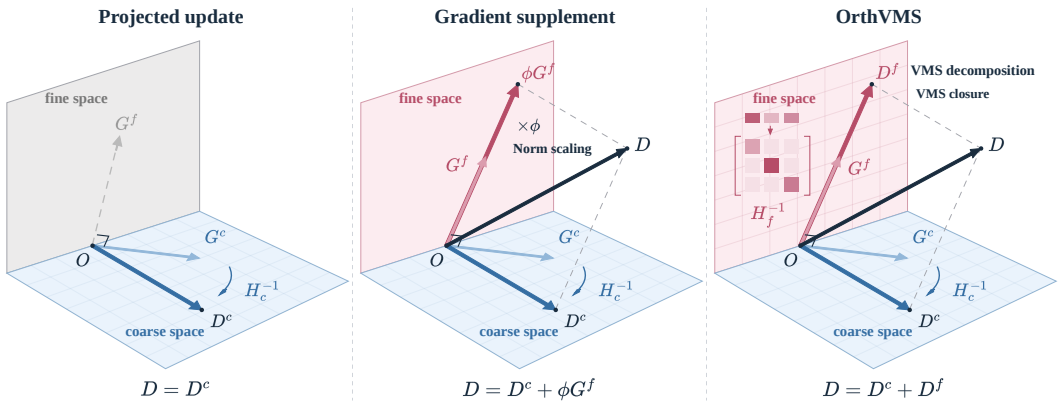

Figure 1: Updates in the projected coarse space and its orthogonal fine space. *Left.* Projected update preconditions the coarse gradient G^c and discards the fine gradient G^f. *Middle.* Gradient supplement adds G^f back with a heuristic norm scale ϕ borrowed from the coarse space. *Right.* OrthVMS builds a diagonal fine preconditioner H_f^{-1} from the residual energy history (rose squares) and adds the preconditioned fine update D^f to the coarse update D^c. Arrows in the fine space show one residual column, and H_c^{-1} denotes the coarse preconditioner.

To retain full-parameter training, GaLore (Zhao et al., 2024) shifts low-rank compression from weight increments to gradients, maintaining compact optimizer states in a projected subspace. APOLLO (Zhu et al., 2025) estimates channel-wise or tensor-wise scaling from low-rank auxiliary states and applies it to the full gradient. FiRA (Chen et al., 2026) keeps the projected adaptive update and rescales the orthogonal residual by a norm ratio from the projected space, heuristically compensating for the gradient discarded by projection. *However, existing approaches do not fully exploit the orthogonal complement of the projected subspace.*

Table 1 summarizes the three update schemes illustrated in Figure 1. From the perspective of preconditioning, these methods share a common concern, namely how to construct a suitable metric for gradient updates at an affordable cost. Newton's method provides a classical reference by correcting the gradient for curvature with the inverse Hessian. In LLM training, however, computing and storing the full Hessian and solving the associated linear system is usually prohibitive. Practical optimizers therefore adopt tractable preconditioners built from gradient history or matrix structure. AdamW (Loshchilov & Hutter, 2017) forms a diagonal adaptive metric from the second moment of gradients and scales the first-order momentum by its inverse.

For memory efficiency, projection-based optimizers restrict adaptive preconditioning to a reduced subspace. As shown in the left panel of Figure 1, GaLore (Zhao et al., 2024) projects the gradient onto a compact subspace, maintains first- and second-moment states there, and maps the preconditioned update back to the original space. The gradient component in the orthogonal complement is thus discarded. APOLLO (Zhu et al., 2025) applies adaptive scaling to the full gradient, but estimates this scaling only from the projected gradient and thus ignores information in the orthogonal complement. FiRA (Chen et al., 2026), shown in the middle panel, restores the orthogonal residual but builds no adaptive preconditioner for it. Instead, it rescales the residual by a norm ratio borrowed from the projected space. This heuristic compensation offers no descent guarantee for the residual update and no convergence guarantee for the overall optimization process.

This naturally leads to the central question of our work:

Can we construct a full-space preconditioner that fully exploits information from both the projected space and its orthogonal complement while remaining memory efficient?

To address this question, we draw on Variational Multiscale (VMS) modeling from computational mechanics (Hughes, 1995, Hughes et al., 1998). Classical VMS decomposes a solution into resolved coarse scales and unresolved fine scales, and it captures fine-scale effects through a residual-driven

closure. We regard the projected space and its orthogonal complement as the coarse and fine spaces, respectively, and model their preconditioning in a decoupled manner. Building on this idea, we propose **OrthVMS**, an **Orth**ogonal **V**ariational **M**ulti**S**cale correction for memory-efficient training.

As illustrated in the right panel of Figure 1, OrthVMS builds a full-space preconditioner from two decoupled components on the coarse and fine spaces. The coarse component uses compact adaptive moments to produce a low-memory update. The fine component preconditions the orthogonal gradient residual with its energy history, yielding a residual-driven correction. Since the fine preconditioning preserves orthogonality to the coarse space, the two updates remain strictly orthogonal. Together, they extend adaptive updates to the entire parameter space while requiring only one additional persistent vector per projected weight matrix. We further establish a convergence guarantee for the joint coarse and fine updates under explicit assumptions.

Our main contributions are as follows:

- **A cross-disciplinary perspective.** We bring the idea of variational multiscale modeling from computational mechanics into LLM optimizer design. By treating the projected space and its orthogonal complement as coarse and fine spaces, we connect scale separation and residual-driven closure with memory-efficient preconditioning.
- **The OrthVMS framework with theoretical guarantees.** We build a full-space preconditioner from decoupled coarse and fine components that adds only vector-sized optimizer states, and our convergence analysis provides a theoretical guarantee for their joint update.
- **Extensive experimental validation.** Extensive experiments on LLM pretraining validate the effectiveness of OrthVMS and demonstrate its advantages over existing memory-efficient training methods.

2 Related Work

Parameter-efficient adaptation. Parameter-efficient fine-tuning reduces adaptation costs by limiting trainable parameters. LoRA (Hu et al., 2021) freezes pretrained weights and learns low-rank increments, while QLoRA (Dettmers et al., 2023) further quantizes the frozen weights. ReLoRA (Lialin et al., 2024) supports pretraining by periodically merging and restarting low-rank updates. OrthVMS directly updates the original weights while compressing optimizer states.

Memory-efficient optimizers. Adafactor (Shazeer & Stern, 2018) factorizes second moments into row and column statistics. Adam-mini (Zhang et al., 2025) shares adaptive learning rates within parameter blocks while keeping per-parameter momentum. GaLore (Zhao et al., 2024) maintains adaptive moments of low-rank projected gradients and maps compact updates back to the full space, enabling full-parameter training. APOLLO (Zhu et al., 2025) applies channel-wise or tensor-wise scaling estimated from compact statistics to the original gradient. FiRA (Chen et al., 2026) adds back the orthogonal residual, scaled by norm ratios from the projected space. In contrast, OrthVMS preconditions this residual using its own energy history.

Variational multiscale modeling. VMS separates resolved and unresolved scales and models fine-scale responses through residual closure (Hughes, 1995, Hughes et al., 1998). Its application to large eddy simulation illustrates the selective modeling of fine-scale effects (Hughes et al., 2000). OrthVMS introduces the VMS perspective to LLM optimization, treating the projected subspace and its orthogonal complement as the coarse and fine spaces, respectively. Its scalar preconditioning and compact energy history are modeling choices tailored to the optimizer-memory budget. More related work is discussed in Appendix A.

3 Preliminaries

We first introduce a preconditioning perspective on LLM optimizers, then describe low-rank projection, the orthogonal decomposition, and the variational multiscale viewpoint used by OrthVMS.

3.1 A PRECONDITIONING PERSPECTIVE ON LLM OPTIMIZERS

For LLM training, computing and storing the full Hessian and solving the associated linear system are prohibitively expensive. Let $x_t \in \mathbb{R}^d$ denote the parameter vector and f the objective function. Newton's method takes the form

$$x_{t+1} = x_t - H_t^{-1} \nabla f(x_t), \qquad H_t = \nabla^2 f(x_t). \tag{1}$$

Practical optimizers reduce the cost of the Newton step in Eq. (1) through approximate preconditioning. AdamW (Loshchilov & Hutter, 2017) constructs a diagonal adaptive metric from gradient second moments and applies its inverse to the first moment.

For a trainable weight matrix $W_t \in \mathbb{R}^{m \times n}$, write $x_t = \mathrm{vec}(W_t)$, where vec stacks columns, and let $G_t = \nabla_W f(W_t)$ be its gradient computed by backpropagation. With first- and second-moment matrices $M_t, V_t \in \mathbb{R}^{m \times n}$, this metric is $\widehat{H}_t = \mathrm{Diag}(\mathrm{vec}(\sqrt{V_t}))$, where Diag forms a diagonal matrix. Its inverse gives $\widehat{H}_t^{-1} \mathrm{vec}(M_t) = \mathrm{vec}(\frac{M_t}{\sqrt{V_t}})$, yielding the gradient update $W_{t+1} = W_t - \eta_t(\frac{M_t}{\sqrt{V_t}})$.

Matrix-based optimizers offer other forms of preconditioning: Muon (Jordan et al., 2024) approximately orthogonalizes the momentum matrix using Newton–Schulz iterations, while RMNP (Deng et al., 2026) replaces these iterations with row-wise ℓ_2 normalization of the momentum matrix.

We use $\langle A, B \rangle_F = \mathrm{tr}(A^\top B)$ and $\|A\|_F$ for the Frobenius inner product and norm. The symbol $\odot$ denotes elementwise multiplication.

3.2 LOW-RANK PROJECTION

Low-rank projection reduces the state required for adaptive preconditioning by maintaining moment estimates in compact coordinates. We present left projection throughout. Right projection is analogous.

Let $P_t \in \mathbb{R}^{m \times r}$ have orthonormal columns, with $P_t^\top P_t = I_r$ and $r < \min(m, n)$. GaLore (Zhao et al., 2024) periodically refreshes P_t using leading left singular vectors of the gradient and computes the compact gradient $R_t = P_t^\top G_t \in \mathbb{R}^{r \times n}$. With decay factors $\beta_1, \beta_2 \in [0, 1)$, its compact moment states satisfy

$$\begin{aligned} M_t^c &= \beta_1 M_{t-1}^c + (1 - \beta_1) R_t, \\ V_t^c &= \beta_2 V_{t-1}^c + (1 - \beta_2)(R_t \odot R_t). \end{aligned} \tag{2}$$

Applying the same construction in projected coordinates gives $\widehat{H}_t^c = \mathrm{Diag}(\mathrm{vec}(\sqrt{V_t^c}))$. The compact preconditioned direction satisfies $\mathrm{vec}(N_t) = (\widehat{H}_t^c)^{-1} \mathrm{vec}(M_t^c)$, equivalently $N_t = \frac{M_t^c}{\sqrt{V_t^c}}$.

The persistent moment and basis storage comprises $2rn$ and mr scalars, respectively. At each step, the lifted direction $P_t N_t$ lies in the current subspace. Basis refreshes allow updates to accumulate across different subspaces, and the compact moments collect gradients expressed in these successive bases.

3.3 ORTHOGONAL COARSE AND FINE SCALES

The left basis induces projectors $\Pi_t^c = P_t P_t^\top$ and $\Pi_t^f = I_m - \Pi_t^c$. They define matrix subspaces

$$\mathcal{C}_t = \{P_t A \mid A \in \mathbb{R}^{r \times n}\}, \qquad \mathcal{F}_t = \{B \in \mathbb{R}^{m \times n} \mid P_t^\top B = 0\}. \tag{3}$$

Under the Frobenius inner product, $\mathbb{R}^{m \times n} = \mathcal{C}_t \oplus \mathcal{F}_t$ is an orthogonal decomposition. The gradient splits exactly as $G_t = G_t^c + G_t^f$, with $G_t^c = P_t R_t$ and $G_t^f = G_t - P_t R_t$. Consequently, $P_t^\top G_t^f = 0$, $\langle G_t^c, G_t^f \rangle_F = 0$, and $\|G_t\|_F^2 = \|G_t^c\|_F^2 + \|G_t^f\|_F^2$.

The current gradient and projection suffice to compute the fine-scale residual G_t^f. Since $P_t N_t \in \mathcal{C}_t$, any correction in $\mathcal{F}_t$ is also orthogonal to the coarse adaptive update. This geometry determines the correction space, and Section 4 specifies its scale.

3.4 Variational Multiscale Viewpoint

Classical VMS (Hughes, 1995, Hughes et al., 1998) separates a boundary-value problem into resolved and unresolved scales. Let $\mathcal{V}$ be a real Hilbert space of admissible fields and $\mathcal{L} : \mathcal{V} \to \mathcal{V}^*$ a bounded linear operator, where $\mathcal{V}^*$ is the dual space. The bilinear form $a(w, u) = (\mathcal{L}u)(w)$ and load functional $b \in \mathcal{V}^*$ define the weak problem $a(w, u) = b(w)$ for all test functions $w \in \mathcal{V}$.

A decomposition into closed coarse and fine subspaces $\mathcal{V} = \mathcal{V}_c \oplus \mathcal{V}_f$ splits both the solution $u = u_c + u_f$ and test functions $w = w_c + w_f$. Testing in $\mathcal{V}_c$ gives the coarse balance $a(w_c, u_c) + a(w_c, u_f) = b(w_c)$. Testing in $\mathcal{V}_f$ gives $a(w_f, u_f) = r_f(w_f)$, where $r_f(w_f) = b(w_f) - a(w_f, u_c)$. Thus, the unresolved response is driven by the coarse residual restricted to fine-scale tests. When the fine problem is well posed, solving for u_f and substituting it into the coarse balance closes the coarse equation. The term $a(w_c, u_f)$ carries the unresolved-scale feedback.

VMS closure approximates the fine-scale response. A scalar closure replaces the restricted bilinear form with $\sigma\langle w_f, u_f\rangle$ for $\sigma > 0$, giving the modeled equation

$$\sigma\langle w_f, u_f\rangle = r_f(w_f) \qquad \text{for all } w_f \in \mathcal{V}_f. \tag{4}$$

OrthVMS transfers this construction to LLM optimization using the orthogonal spaces in Eq. (3). For column j, $\mathcal{V}_f$ corresponds to $\mathrm{range}(\Pi_t^f)$ and u_f to the fine response $b_{t,j}$. We model the residual functional by $r_f(z) = \langle z, G_t^f[:, j]\rangle$. The next section constructs the column scales $\sigma_{t,j}$ for Eq. (4).

4 Method

OrthVMS introduces the VMS perspective from computational mechanics to LLM optimization, treating the projected subspace and its orthogonal complement as the coarse and fine spaces, respectively. The coarse update uses compact adaptive moments, while the fine response is driven by the orthogonal residual and scaled using its own energy history.

Algorithm 1 OrthVMS with left projection

Input W_1, rank r, refresh interval T_{p}, steps T
Parameters η_t, $\beta_1, \beta_2, \beta_3$, α_c, τ_t
$M_0^c, V_0^c \leftarrow 0$, $s_0 \leftarrow 0$
for $t = 1, \ldots, T$ **do**
  $G_t \leftarrow \nabla_W f_t(W_t)$
  if $t = 1$ or $(t-1) \bmod T_{\mathrm{p}} = 0$ **then**
    $U_t \Sigma_t V_t^\top \leftarrow \mathrm{SVD}(G_t)$
    $P_t \leftarrow U_t[:, 1\!:\!r]$
  else
    $P_t \leftarrow P_{t-1}$
  end if
  $R_t \leftarrow P_t^\top G_t$ *coarse update*
  Update M_t^c, V_t^c using Eq. (2)
  $N_t \leftarrow \frac{M_t^c}{\sqrt{V_t^c}}$, $\bar{\eta}_t \leftarrow \eta_t \frac{\sqrt{1-\beta_2^t}}{1-\beta_1^t}$
  $D_t^c \leftarrow \alpha_c P_t N_t$, $G_t^f \leftarrow G_t - P_t R_t$
  for $j = 1, \ldots, n$ **do** *fine update*
    Update $s_{t,j}, \sigma_{t,j}$ using Eq. (7)
    $B_t[:, j] \leftarrow G_t^f[:, j]/\sigma_{t,j}$
  end for
  $D_t^f \leftarrow \tau_t \alpha_c \sqrt{\frac{r}{m}}\, B_t$
  $W_{t+1} \leftarrow W_t - \bar{\eta}_t (D_t^c + D_t^f)$
end for
return W_{T+1}

4.1 Coarse–Fine Preconditioning

To maintain memory efficiency while preconditioning the fine space, OrthVMS constructs a full-space preconditioner from two decoupled components on the coarse and fine spaces. For column j, the second moments in Eq. (2) define the coarse block $\widehat{H}_{t,j}^c = \mathrm{diag}(\sqrt{V_t^c[:, j]})$. The fine block $\widehat{H}_{t,j}^f = \sigma_{t,j} I_{m-r}$ uses one shared scale within that column's fine space.

Choose $Q_t \in \mathbb{R}^{m \times (m-r)}$ so that $[P_t, Q_t]$ is orthogonal.

The column metric and its block representation are

$$\widehat{H}_{t,j} = P_t \widehat{H}_{t,j}^c P_t^\top + \sigma_{t,j} \Pi_t^f, \qquad [P_t, Q_t]^\top \widehat{H}_{t,j} [P_t, Q_t] = \begin{pmatrix} \widehat{H}_{t,j}^c & 0 \\ 0 & \widehat{H}_{t,j}^f \end{pmatrix}. \tag{5}$$

For $\sigma_{t,j} > 0$, combining the coarse inverse with the fine inverse yields

$$\widehat{H}_{t,j}^{-1} = P_t \big(\widehat{H}_{t,j}^c\big)^{-1} P_t^\top + \frac{1}{\sigma_{t,j}} \Pi_t^f. \tag{6}$$

Applying the coarse inverse in Eq. (6) column by column to the lifted moment $P_t M_t^c$ gives $P_t N_t$, where $N_t = \frac{M_t^c}{\sqrt{V_t^c}}$. The coarse update is $D_t^c = \alpha_c P_t N_t$.

On the fine space, the metric acts as multiplication by $\sigma_{t,j}$, giving the scalar closure in Section 3.4, Eq. (4). Using the current residual $G_t^f = G_t - P_t R_t$ as the driving term yields $b_{t,j} = \frac{G_t^f[:,j]}{\sigma_{t,j}}$. With $S_t = \mathrm{diag}(\sigma_{t,1}, \ldots, \sigma_{t,n})$, the matrix response is $B_t = G_t^f S_t^{-1}$. One energy statistic per column determines the fine scale, adding only n persistent scalars. Section 4.2 gives the corresponding energy update.

4.2 Preconditioning from Fine-Scale Energy

To specify the fine block in Eq. (5), we estimate $\sigma_{t,j}$ from the column's smoothed residual energy. For $\beta_3 \in [0, 1)$, update

$$\begin{aligned} e_{t,j} &= \frac{\|G_t^f[:,j]\|_2^2}{m-r}, \quad & s_{t,j} &= \beta_3 s_{t-1,j} + (1-\beta_3) e_{t,j}, \\ \widehat{s}_{t,j} &= \frac{s_{t,j}}{1-\beta_3^t}, & \sigma_{t,j} &= \sqrt{\widehat{s}_{t,j}}. \end{aligned} \tag{7}$$

Here $m-r$ is the fine-space dimension. For any such complement basis Q_t, $\|G_t^f[:,j]\|_2^2 = \|Q_t^\top G_t[:,j]\|_2^2$. Thus, $e_{t,j}$ is the mean squared coordinate magnitude in the fine space, independent of the choice of Q_t. The moving average $s_{t,j}$ smooths this energy over time, and $\widehat{s}_{t,j}$ corrects its initialization bias.

Using the scales from Eq. (7) in $B_t = G_t^f S_t^{-1}$ preconditions each current residual column according to its own second-moment history. FiRA (Chen et al., 2026) obtains its residual multiplier from the norm ratio between the projected adaptive update and the projected gradient.

Since S_t^{-1} multiplies each column by a scalar, $P_t^\top B_t = (P_t^\top G_t^f) S_t^{-1} = 0$. Thus, B_t remains in the fine space and is orthogonal to the coarse update, giving $\langle D_t^c, B_t \rangle_F = 0$.

4.3 Combining Coarse and Fine Updates

To balance the fine response B_t with the coarse update D_t^c, we set $D_t^f = \alpha_{f,t} B_t$, where $\alpha_{f,t} = \tau_t \alpha_c \sqrt{\frac{r}{m}}$ and $\tau_t \geq 0$. We set $\tau_t = 1$ by default. Under a unit-coordinate-energy model, N_t and B_t have squared norms of order rn and $(m-r)n$, respectively. The factor $\sqrt{\frac{r}{m}}$ compensates for this dimensional imbalance in the low-rank regime, while τ_t controls the relative fine strength.

Combining the two contributions gives

$$D_t = D_t^c + D_t^f, \qquad W_{t+1} = W_t - \bar{\eta}_t D_t, \tag{8}$$

where the bias-corrected step size $\bar{\eta}_t$ in Algorithm 1 multiplies both contributions. The orthogonality established in Section 4.2 gives $\|D_t\|_F^2 = \|D_t^c\|_F^2 + \|D_t^f\|_F^2$. With $\tau_t = 0$, the update reduces to the coarse adaptive step. A nonzero D_t^f extends the update beyond the coarse space.

4.4 Algorithm and Projection Refresh

Algorithm 1 starts each iteration by computing G_t. At the first step and every T_{p} steps thereafter, it constructs P_t from the leading left singular vectors of G_t. Between refreshes, it reuses the previous basis. The compact moments and fine energy vector keep their stored entries across refreshes.

Using the current basis, the algorithm projects G_t to R_t and updates M_t^c, V_t^c through Eq. (2). These moments give the preconditioned direction N_t, which is lifted to the coarse update D_t^c. The residual $G_t^f = G_t - P_t R_t$ then supplies the fine path. For each column, Eq. (7) updates the energy history and scale $\sigma_{t,j}$. Scaling the residual columns by their inverse scales forms B_t, which is multiplied by $\tau_t \alpha_c \sqrt{\frac{r}{m}}$ to obtain D_t^f. Finally, Eq. (8) combines the separately preconditioned coarse and fine updates D_t^c and D_t^f and applies the shared step size $\bar{\eta}_t$.

4.5 Memory and Computational Cost

OrthVMS stores the coarse moments $M_t^c, V_t^c \in \mathbb{R}^{r\times n}$ and projection basis $P_t \in \mathbb{R}^{m\times r}$, requiring $2rn$ and mr scalars, respectively. Fine-space preconditioning uses one energy statistic $s_{t,j}$ per column, adding only n persistent scalars.

The coarse update and basis refresh have the same computational cost as GaLore. The additional computation comes from the fine response. Constructing $G_t^f = G_t - P_t R_t$ requires an extra matrix product $P_t R_t$, costing $O(mnr)$. Computing the column energies, updating $s_{t,j}$, and forming D_t^f then require $O(mn)$ work per step.

5 Convergence Analysis

We analyze the convergence behavior of OrthVMS in terms of the average squared gradient norm. The resulting bound accounts for preconditioning in both spaces and changes in the projection basis during training.

For one matrix block $W_t \in \mathbb{R}^{m\times n}$ with the remaining parameters fixed, write $\mathsf{g}_t = \nabla f(W_t)$ and $G_t = \mathsf{g}_t + \xi_t$, and let b denote the fixed minibatch size over T updates. For the update in Eq. (8), the effective step size and fine weight are $h_t = \bar{\eta}_t$ and $\gamma_t = \tau_t \alpha_c \sqrt{\frac{r}{m}}$.

Our conditions are stated as Assumptions 1 through 6 in Appendix B.1. Proposition B.1 and Lemma B.1 show how a positive fine weight makes the effective preconditioner control the full gradient. Define the accumulated basis variation by $\mathcal{V}_T = \sum_{t=2}^{T} \mathbb{E}\|P_t - P_{t-1}\|_2^2$, and let $f_* = \inf_W f(W)$ and $\Delta = f(W_1) - f_*$. The spectral constants μ, ν and error constants below are given explicitly in Eqs. (B.13) and (B.58).

Theorem 1 (Finite horizon stationarity)**.** *Under Assumptions 1 through 6 and the algorithm conditions in Appendix B.1, consider the stabilized update of Algorithm 1 defined in Appendix B.2, with constant effective step size* $0 < h \le (2L\nu)^{-1}$*. Then, for every integer* $T \ge 1$*,*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\|\nabla f(W_t)\|_F^2 \le \frac{2\Delta}{\mu h T} + \frac{C_{\text{init}}}{\mu^2 T} + \frac{C_{\text{drift}} h^2}{\mu^2} + \frac{C_{\text{basis}} \mathcal{V}_T}{\mu^2 T} + \frac{C_{\text{noise}} \upsilon^2}{\mu^2 \mathsf{b}}. \tag{9}$$

For a fixed step size, the optimization and initialization terms in Theorem 1 decay as $O(T^{-1})$. The remaining terms scale as $O(h^2 + \mathcal{V}_T/T + \upsilon^2/\mathsf{b})$. The proof is given in Appendix B.4.

6 Experiments

We evaluate OrthVMS on two pre-training benchmarks, LLaMA (Llama, 2023) on C4 (Raffel et al., 2020) and gated linear attention (GLA) (Yang et al., 2023) on FineWeb-Edu (Penedo et al., 2024), and compare it with AdamW (Loshchilov & Hutter, 2017) and memory-efficient optimizers GaLore (Zhao et al., 2024), FiRA (Chen et al., 2026), APOLLO, and APOLLO-Mini (Zhu et al., 2025). Following the evaluation practice of APOLLO and GaLore, we report final validation perplexity (PPL) and memory to assess model quality and memory cost. In Tables 2, 3, and 4, **bold** and cyan mark the best and second-best results, respectively, and the OrthVMS row is highlighted in alice blue.

6.1 Experimental Setup

Models and training. The LLaMA experiments cover 60M, 130M, and 350M models, while the GLA experiments cover these scales and extend to 1B. All methods use the same sequence lengths and training steps at each model scale within each benchmark. Additional experimental settings and result-selection rules are provided in Appendix C.

Metrics. Lower PPL indicates better validation quality. Memory denotes the per-GPU persistent state of model weights, optimizer states, cached projection bases, and states of non-projected parameters, reported in GB (10^9 bytes). The main tables summarize each method's best final PPL among the reported configurations.

Memory measurement. We measure memory after state initialization in separate short runs. LLaMA measurements use eight A100-80GB GPUs with standard DDP and BF16 weights and moments. GLA measurements use a single NVIDIA H200 GPU with FP32 weights and optimizer states. Memory comparisons use the same precision settings within each benchmark.

6.2 LLaMA on C4 with Short and Intermediate-Length Sequences

Table 2: Final PPL (↓) and memory (GB) for LLaMA pre-training on C4.

	60M		130M		350M	
Method	PPL ↓	Memory	PPL ↓	Memory	PPL ↓	Memory
AdamW	33.32	0.348	24.78	0.805	23.78	2.208
GaLore	34.13	0.280	24.92	0.611	23.77	1.389
APOLLO	31.05	0.280	22.84	0.611	20.17	1.389
APOLLO-Mini	31.50	0.248	23.49	0.465	20.76	1.000
FiRA	30.36	0.280	22.78	0.611	21.65	1.389
OrthVMS (Ours)	**29.95**	0.280	**22.62**	0.611	**19.45**	1.389

Setup. Following the pre-training setup of APOLLO (Zhu et al., 2025) and GaLore (Zhao et al., 2024), we train 60M, 130M, and 350M LLaMA models on C4 using eight NVIDIA A100 GPUs. The 60M & 130M models use 256-token sequences, and the 350M model uses 1024-token sequences. Training lasts 10K, 20K, and 60K steps, respectively.

Results. As shown in Table 2, OrthVMS reduces PPL relative to AdamW by approximately 3.37, 2.16, and 4.34 at 60M, 130M, and 350M. Compared with the other memory-efficient optimizers, it achieves the lowest PPL at all three scales, improving over the strongest baseline at each scale by approximately 0.41, 0.16, and 0.73. Meanwhile, OrthVMS adds only 0.099, 0.221, and 0.591 MB of persistent state relative to GaLore. The left two panels of Figure 2 show the corresponding training loss and validation PPL trajectories at 350M.

Zero-shot evaluation. To assess downstream performance, we evaluate the LLaMA-350M checkpoints after 60K pre-training steps on ten zero-shot tasks covering commonsense and mathematical reasoning (Zhu et al., 2025). Table 3 reports the results. The evaluation splits and metrics are specified in Appendix C.1. OrthVMS achieves the highest average score, 43.94%, compared with 43.75% for APOLLO, and leads on five of the ten tasks. Performance varies across tasks: APOLLO performs best on BoolQ, WinoGrande, and OpenBookQA, FiRA on RTE, and APOLLO-Mini on PIQA.

Table 3: Zero-shot performance on commonsense and mathematical reasoning tasks for LLaMA-350M models trained on C4 for 60K steps with sequence length 1024. Task scores and Average are percentages (↑).

Method	Memory (GB)	BoolQ	RTE	HS	WG	OBQA	ARC-E	ARC-C	PIQA	SciQ	MathQA	Average
AdamW	2.208	60.00	52.35	29.55	52.25	28.60	36.70	22.87	61.37	57.80	21.31	42.28
APOLLO	1.389	**62.17**	52.35	32.52	**53.35**	**29.60**	37.50	21.59	64.53	62.80	21.14	43.75
APOLLO-Mini	1.000	61.31	51.26	31.92	50.28	29.00	38.85	23.04	**65.29**	62.70	20.94	43.46
FiRA	1.389	61.44	**54.51**	31.33	51.07	27.00	37.33	22.01	62.95	61.70	20.80	43.01
OrthVMS (Ours)	1.389	58.13	54.15	**33.26**	51.46	28.00	**39.60**	**23.72**	63.87	**65.60**	**21.57**	**43.94**

6.3 GLA on FineWeb-Edu with Long Sequences

Setup. We train 60M, 130M, 350M, and 1B GLA models on FineWeb-Edu using a single NVIDIA H200 GPU for 10K, 20K, 60K, and 100K steps, respectively. Our setup draws on OmniOpt (Li et al., 2026). All scales use a context-segment limit of 4096 tokens, with multiple segments packed into 32,768-token inputs. Appendix C.2 details the model configurations and result-selection rules.

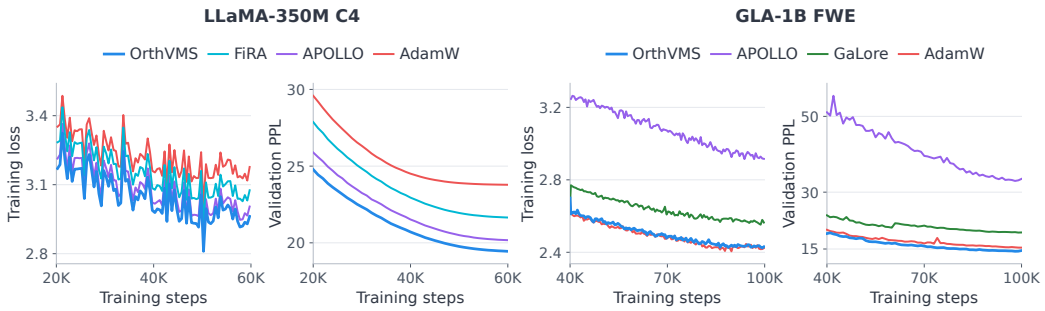

Figure 2: Training loss and validation PPL for selected optimizers. Left two panels: LLaMA-350M on C4, starting at 20K steps. Right two panels: GLA-1B on FineWeb-Edu, starting at 40K steps. Training loss is averaged within 500-step bins, and PPL is shown at recorded validation steps.

Results. As shown in Table 4, OrthVMS reduces PPL relative to AdamW by approximately 2.35, 1.65, and 0.78 at 60M, 130M, and 1B, and has a PPL approximately 0.54 higher at 350M. At 1B, it also reduces memory by approximately 32.5%. Compared with the other memory-efficient optimizers, OrthVMS achieves the lowest PPL at all four scales under the reported configurations. Its memory remains close to GaLore across all scales, with only about 2.86 MB of additional persistent state at 1B. The right two panels of Figure 2 show the 1B trajectories from 40K steps onward.

Table 4: Final validation PPL and memory (GB) for GLA pre-training on FineWeb-Edu. The context-segment limit is 4096 tokens and the packed input length is 32,768 tokens. The four scales train for 10K, 20K, 60K, and 100K steps.

	60M		130M		350M		1B	
Method	PPL ↓	Memory	PPL ↓	Memory	PPL ↓	Memory	PPL ↓	Memory
AdamW	49.88	0.722	31.74	1.612	**20.25**	4.494	15.36	16.386
GaLore	105.96	0.422	37.64	1.104	25.17	2.964	19.37	11.053
APOLLO	128.62	0.387	64.37	1.010	38.78	2.551	33.50	9.409
APOLLO-Mini	98.40	0.242	49.65	0.539	22.27	1.503	19.14	5.471
FiRA	86.56	0.422	47.41	1.104	28.53	2.964	29.96	11.053
OrthVMS (Ours)	**47.53**	0.423	**30.09**	1.105	20.79	2.965	**14.59**	11.056

7 Limitations and Future Work

OrthVMS periodically refreshes the projection basis using SVD, adding computational overhead for large weight matrices. Future work will explore cheaper subspace updates. Our current evaluation focuses on Transformer-based language models. We plan to extend OrthVMS to multimodal and diffusion models and evaluate it on a wider range of benchmarks.

8 Conclusion

We introduce OrthVMS, a memory-efficient optimizer for LLM training inspired by VMS in computational mechanics. Treating the projected subspace and its orthogonal complement as coarse and fine spaces, OrthVMS combines compact adaptive moments with a residual-driven fine response preconditioned by its own energy history. It adds only a vector of persistent states and admits a convergence guarantee under explicit assumptions. Extensive experiments across model architectures, scales, and sequence lengths show that OrthVMS consistently outperforms existing memory-efficient optimizers. This VMS viewpoint bridges classical multiscale theory and modern AI optimization, opening new directions for principled memory-efficient optimizer design.

AI USE STATEMENT

In this work, we used generative AI tools for translation, code implementation, and experimental design. We have not used generative AI tools for synthetic data generation or dataset cleaning, and qualitative analyses are not applicable to this work. Additionally, we used generative AI tools for literature searches, manuscript drafting and editing, LaTeX formatting, code debugging, and visualization. We have reviewed all AI-assisted work. We checked and revised the text, mathematics, code, and experimental analyses. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Rohan Anil, Vineet Gupta, Tomer Koren, and Yoram Singer. Memory efficient adaptive optimization. *Advances in Neural Information Processing Systems*, 32, 2019.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

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? *Advances in Neural Information Processing Systems*, 38:120680–120712, 2026.

Shenyang Deng, Zhuoli Ouyang, Tianyu Pang, Zihang Liu, Ruochen Jin, Shuhua Yu, and Yaoqing Yang. Rmnp: Row-momentum normalized preconditioning for scalable matrix-based optimization. *arXiv preprint arXiv:2603.20527*, 2026.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization. *arXiv preprint arXiv:2110.02861*, 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115, 2023.

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

Andi Han, Jiaxiang Li, Wei Huang, Mingyi Hong, Akiko Takeda, Pratik Jawanpuria, and Bamdev Mishra. Sltrain: a sparse plus low rank approach for parameter and memory efficient pretraining. *Advances in Neural Information Processing Systems*, 37:118267–118295, 2024.

Yongchang Hao, Yanshuai Cao, and Lili Mou. Flora: Low-rank adapters are secretly gradient compressors. *arXiv preprint arXiv:2402.03293*, 2024.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. Lora+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*, 2024.

Yutong He, Pengrui Li, Yipeng Hu, Chuyan Chen, and Kun Yuan. Subspace optimization for large language models with convergence guarantees. *arXiv preprint arXiv:2410.11289*, 2024.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

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. *arXiv preprint arXiv:2106.09685*, 2021.

Thomas JR Hughes. Multiscale phenomena: Green's functions, the dirichlet-to-neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Computer methods in applied mechanics and engineering*, 127(1-4):387–401, 1995.

Thomas JR Hughes, Gonzalo R Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale method—a paradigm for computational mechanics. *Computer methods in applied mechanics and engineering*, 166(1-2):3–24, 1998.

Thomas JR Hughes, Luca Mazzei, and Kenneth E Jansen. Large eddy simulation and the variational multiscale method. *Computing and visualization in science*, 3(1):47–59, 2000.

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*, 6(3):4, 2024.

Dawid Kopiczko, Tijmen Blankevoort, and Yuki Asano. Vera: Vector-based random matrix adaptation. In *International Conference on Learning Representations*, volume 2024, pp. 6815–6835, 2024.

Siyuan Li, Jiabao Pan, Yumou Liu, Zhuoli Ouyang, Xin Jin, Xinglong Xu, Jingxuan Wei, Shengye Pang, Jintao Che, Xuanhe Zhou, et al. Omniopt: Taxonomy, geometry, and benchmarking of modern optimizers. *arXiv preprint arXiv:2607.04033*, 2026.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)*, pp. 4582–4597, 2021.

Vladislav Lialin, Sherin Muckatira, Namrata Shivagunde, and Anna Rumshisky. Relora: High-rank training through low-rank updates. In *International Conference on Learning Representations*, volume 2024, pp. 49405–49421, 2024.

Hong Liu, Zhiyuan Li, David Hall, Percy Liang, and Tengyu Ma. Sophia: A scalable stochastic second-order optimizer for language model pre-training. In *International conference on learning representations*, volume 2024, pp. 1621–1650, 2024a.

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. *arXiv preprint arXiv:2402.09353*, 2024b.

Hugo Touvron Llama. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2407.21783*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Qijun Luo, Hengxu Yu, and Xiao Li. Badam: A memory efficient full parameter optimization method for large language models. *Advances in Neural Information Processing Systems*, 37: 24926–24958, 2024.

Yang Luo, Xiaozhe Ren, Zangwei Zheng, Zhuo Jiang, Xin Jiang, and Yang You. Came: Confidence-guided adaptive memory efficient optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4442–4453, 2023.

Kai Lv, Yuqing Yang, Tengxiao Liu, Qipeng Guo, and Xipeng Qiu. Full parameter fine-tuning for large language models with limited resources. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8187–8198, 2024.

James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature. In *International conference on machine learning*, pp. 2408–2417. PMLR, 2015.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37:121038–121072, 2024.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024.

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.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: international conference for high performance computing, networking, storage and analysis*, pp. 1–16. IEEE, 2020.

Jie Ren, Samyam Rajbhandari, Reza Yazdani Aminabadi, Olatunji Ruwase, Shuangyan Yang, Minjia Zhang, Dong Li, and Yuxiong He. {Zero-offload}: Democratizing {billion-scale} model training. In *2021 USENIX Annual Technical Conference (USENIX ATC 21)*, pp. 551–564, 2021.

Thomas Robert, Mher Safaryan, Ionut-Vlad Modoranu, and Dan Alistarh. Ldadam: Adaptive optimization from low-dimensional gradient statistics. In *International Conference on Learning Representations*, volume 2025, pp. 4778–4814, 2025.

Noam Shazeer and Mitchell Stern. Adafactor: Adaptive learning rates with sublinear memory cost. In *International conference on machine learning*, pp. 4596–4604. PMLR, 2018.

Nikhil Vyas, Depen Morwani, Rosie Zhao, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham Kakade. Soap: Improving and stabilizing shampoo using adam for language modeling. In *International Conference on Learning Representations*, volume 2025, pp. 93423–93444, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023.

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

Zhenyu Zhang, Ajay Jaiswal, Lu Yin, Shiwei Liu, Jiawei Zhao, Yuandong Tian, and Zhangyang Wang. Q-galore: Quantized galore with int4 projection and layer-adaptive low-rank gradients. *arXiv preprint arXiv:2407.08296*, 2024.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. Galore: Memory-efficient llm training by gradient low-rank projection. *arXiv preprint arXiv:2403.03507*, 2024.

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. *Proceedings of Machine Learning and Systems*, 7, 2025.

APPENDIX CONTENTS

A Additional Related Work **14**

B Theoretical Results **15**

B.1 Assumptions 15

B.2 Effective Preconditioner and Spectral Bounds 15

B.3 Direction, Momentum, and Update Errors 18

B.4 Proof of the Main Theorem 24

C Experimental Configurations and Additional Results **26**

C.1 LLaMA C4 Configuration 26

C.2 FineWeb-Edu GLA Configuration 26

A Additional Related Work

Parameter-efficient fine-tuning. Parameter-efficient fine-tuning adapts a pretrained model by training a small set of parameters. Adapters insert small bottleneck modules into each layer (Houlsby et al., 2019), and prefix tuning learns continuous prompts that steer a frozen model (Li & Liang, 2021). LoRA (Hu et al., 2021) learns low-rank weight increments and has inspired many variants. AdaLoRA (Zhang et al., 2023) allocates the rank budget adaptively across weight matrices. DoRA (Liu et al., 2024b) decomposes each weight into magnitude and direction and applies the low-rank update to the direction. LoRA+ (Hayou et al., 2024) assigns different learning rates to the two low-rank factors, VeRA (Kopiczko et al., 2024) shares frozen random matrices across layers and learns only scaling vectors, and PiSSA (Meng et al., 2024) initializes the adapters with the principal singular components of the pretrained weights. QLoRA (Dettmers et al., 2023) further quantizes the frozen weights. These methods restrict the trainable parameters, whereas OrthVMS keeps full-parameter updates and compresses only the optimizer states.

Low-rank and subspace training. Several works bring low-rank structure to pretraining or full-parameter training. ReLoRA (Lialin et al., 2024) periodically merges and restarts low-rank updates, and SLTrain (Han et al., 2024) parameterizes weights as the sum of a sparse and a low-rank component. Flora (Hao et al., 2024) shows that LoRA updates act as random projections of the gradient and resamples the projections to accumulate high-rank updates. GaLore (Zhao et al., 2024) projects gradients onto subspaces obtained from periodic SVD, and Q-GaLore (Zhang et al., 2024) further quantizes the projection matrices and weights. GoLore (He et al., 2024) analyzes the convergence of subspace methods and uses random projections to obtain guarantees under stochastic gradients. LDAdam (Robert et al., 2025) keeps low-dimensional gradient statistics and corrects the projection error with error feedback. FiRA (Chen et al., 2026) compensates the orthogonal residual with a norm-based scaling. OrthVMS shares the projected coarse update with these methods but equips the orthogonal residual with its own preconditioner.

Memory-efficient optimizers. Another line of work reduces optimizer memory without low-rank projection. Adafactor (Shazeer & Stern, 2018) and SM3 (Anil et al., 2019) replace element-wise second moments with factored or cover-based statistics, and CAME (Luo et al., 2023) adds a confidence-guided correction to factored moments. Adam-mini (Zhang et al., 2025) shares learning rates within parameter blocks, and APOLLO (Zhu et al., 2025) estimates channel-wise or tensor-wise scaling from low-rank auxiliary states. 8-bit optimizers (Dettmers et al., 2021) quantize optimizer states with block-wise quantization. BAdam (Luo et al., 2024) updates one block of parameters at a time, and LOMO (Lv et al., 2024) fuses gradient computation with the parameter update to avoid storing full gradients. These techniques are largely orthogonal to the coarse and fine decomposition in OrthVMS.

Structured preconditioning. Second-order and structured optimizers approximate curvature at tractable cost. K-FAC (Martens & Grosse, 2015) approximates the Fisher information with Kronecker factors, and Shampoo (Gupta et al., 2018) maintains Kronecker-factored preconditioners for each weight matrix. SOAP (Vyas et al., 2025) runs Adam in the eigenbasis of the Shampoo preconditioner, and Sophia (Liu et al., 2024a) uses a lightweight diagonal Hessian estimate with clipping. Muon (Jordan et al., 2024) orthogonalizes momentum matrices through Newton-Schulz iterations, and RMNP (Deng et al., 2026) replaces this orthogonalization with row-wise normalization. Most of these methods keep full-size states, whereas OrthVMS targets preconditioning in both spaces under a low-rank state budget.

System-level memory optimization. Memory can also be reduced at the system level. ZeRO (Rajbhandari et al., 2020) partitions optimizer states, gradients, and parameters across data-parallel workers, and ZeRO-Offload (Ren et al., 2021) moves optimizer states and computation to the CPU. Activation checkpointing (Chen et al., 2016) trades recomputation for lower activation storage. These techniques change how memory is managed rather than what the optimizer stores, and they can be combined with OrthVMS.

B THEORETICAL RESULTS

We analyze the convergence of OrthVMS and prove Theorem 1. Under the stated assumptions, the theorem bounds the average squared gradient norm in expectation, providing a convergence guarantee for the combined coarse and fine updates.

We consider left projection for one matrix block $W_t \in \mathbb{R}^{m\times n}$ with the remaining parameters fixed. The projection basis is $P_t \in \mathbb{R}^{m\times r}$ and the fine-space projector is $\Pi_t^f = I_m - P_t P_t^\top$. All matrix inner products and norms are Frobenius unless a subscript specifies otherwise. The norm $\|\cdot\|_2$ denotes the Euclidean norm for vectors and the spectral norm for matrices. The minibatch size b is fixed throughout. The analysis uses exact arithmetic.

B.1 ASSUMPTIONS

Let W_1 be deterministic and let $\mathcal{H}_{t-1}$ be the sigma-algebra describing the history before sampling G_t. The iterate W_t is measurable with respect to $\mathcal{H}_{t-1}$. As in Section 5, write $\mathsf{g}_t = \nabla f(W_t)$ and $\xi_t = G_t - \mathsf{g}_t$.

Assumption 1 (Smoothness)**.** The objective f is differentiable and, for some $L > 0$,

$$\|\nabla f(W) - \nabla f(W')\|_F \le L\|W - W'\|_F \quad \text{for all } W, W'. \tag{B.1}$$

Assumption 2 (Unbiased stochastic gradients)**.** For every iteration,

$$\mathbb{E}[G_t \mid \mathcal{H}_{t-1}] = \mathsf{g}_t, \qquad \mathbb{E}[\xi_t \mid \mathcal{H}_{t-1}] = 0. \tag{B.2}$$

Assumption 3 (Bounded noise variance)**.** For the fixed minibatch size $\mathsf{b} \ge 1$, there is a constant $0 \le \upsilon < \infty$ such that

$$\mathbb{E}[\|\xi_t\|_F^2 \mid \mathcal{H}_{t-1}] \le \frac{\upsilon^2}{\mathsf{b}}. \tag{B.3}$$

Assumption 4 (Lower bounded objective)**.** The value $f_* = \inf_W f(W)$ is finite. We write $\Delta = f(W_1) - f_*$.

Assumption 5 (Bounded stochastic gradients)**.** There is a deterministic constant $0 \le G_{\max} < \infty$ such that $\|G_t\|_F \le G_{\max}$ almost surely at every iteration, including all history used by the moment states.

Assumption 6 (Positive bounded fine weight)**.** There are deterministic constants γ_-, γ_+, independent of the iteration, such that

$$0 < \gamma_- \le \gamma_t = \tau_t \alpha_c \sqrt{\frac{r}{m}} \le \gamma_+ < \infty \quad \text{almost surely.} \tag{B.4}$$

Assumptions 5 and 6 are additional sufficient conditions for the uniform spectral bounds used in this proof. We assume bounded gradient norms separately from the variance bound.

The algorithm conditions are $P_t^\top P_t = I_r$, $1 \le r < \min(m, n)$, $\epsilon > 0$, $\alpha_c > 0$, and fixed $\beta_1, \beta_2, \beta_3 \in [0, 1)$. The moment states start at zero and keep their stored coordinates across basis changes. All iterates and operators are measurable. In particular, P_t may depend on the current stochastic gradient. Dependence between P_t and ξ_t is allowed. Empty sums are interpreted as zero.

B.2 EFFECTIVE PRECONDITIONER AND SPECTRAL BOUNDS

We restore the numerical stabilizer omitted from the main text and Algorithm 1. We use $N_t = M_t^c/(\sqrt{V_t^c} + \epsilon)$ and $B_t[:, j] = G_t^f[:, j]/(\sigma_{t,j} + \epsilon)$, where $\sigma_{t,j} = \sqrt{\widehat{s}_{t,j}}$ and $\epsilon > 0$. All update identities and guarantees below refer to these stabilized expressions.

Using the coarse block $\widehat{H}_{t,j}^c$ from Eq. (5), restore its stabilizer and define the scaled inverse operators by

$$\begin{aligned} \widehat{H}_{t,j}^c + \epsilon I_r &= \operatorname{diag}\big(\sqrt{V_t^c[:, j]} + \epsilon\big), \\ A_{t,j} &= \alpha_c P_t \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} P_t^\top + \frac{\gamma_t}{\sigma_{t,j} + \epsilon} \Pi_t^f, \\ (\mathcal{A}_t X)[:, j] &= A_{t,j} X[:, j]. \end{aligned} \tag{B.5}$$

Here V_t^c follows Eq. (2), and $\sigma_{t,j}$ follows the fine-energy definitions in Section 4.2. In particular, the fine energy is divided by $1-\beta_3^t$ before taking its square root. The effective step $h_t = \bar{\eta}_t$ still multiplies both parts of the update.

Proposition B.1 (Coarse and fine preconditioned energy). *For $\gamma_t \geq 0$, the operator $\mathcal{A}_t$ is symmetric and positive semidefinite. It is positive definite when $\gamma_t > 0$. For every matrix $X \in \mathbb{R}^{m\times n}$,*

$$\begin{aligned}\langle X, \mathcal{A}_t X\rangle_F = \alpha_c \sum_{j=1}^{n} \left\| \left(\widehat{H}_{t,j}^c + \epsilon I_r\right)^{-1/2} P_t^\top X[:,j] \right\|_2^2 \\ + \gamma_t \sum_{j=1}^{n} \frac{\|\Pi_t^f X[:,j]\|_2^2}{\sigma_{t,j}+\epsilon}.\end{aligned} \tag{B.6}$$

Proof. Choose $Q_t \in \mathbb{R}^{m\times(m-r)}$ so that $[P_t, Q_t]$ is orthogonal. The columns of Q_t span the orthogonal complement of the columns of P_t, and

$$\begin{aligned} P_t^\top P_t = I_r, \quad Q_t^\top Q_t = I_{m-r}, \quad P_t^\top Q_t = 0, \\ P_t P_t^\top + Q_t Q_t^\top = I_m, \qquad \Pi_t^f = Q_t Q_t^\top, \quad (\Pi_t^f)^2 = \Pi_t^f. \end{aligned} \tag{B.7}$$

In particular, the coarse and fine operators act on orthogonal spaces. Their coordinate representations are

$$[P_t, Q_t]^\top P_t = \begin{pmatrix} I_r \\ 0 \end{pmatrix}, \qquad [P_t, Q_t]^\top \Pi_t^f [P_t, Q_t] = \begin{pmatrix} 0 & 0 \\ 0 & I_{m-r} \end{pmatrix}. \tag{B.8}$$

Substituting these identities into Eq. (B.5) gives

$$[P_t, Q_t]^\top A_{t,j} [P_t, Q_t] = \begin{pmatrix} \alpha_c\left(\widehat{H}_{t,j}^c + \epsilon I_r\right)^{-1} & 0 \\ 0 & \dfrac{\gamma_t}{\sigma_{t,j}+\epsilon} I_{m-r} \end{pmatrix}. \tag{B.9}$$

The diagonal entries of $\widehat{H}_{t,j}^c + \epsilon I_r$ and the scalar $\sigma_{t,j} + \epsilon$ are positive. Thus both blocks in Eq. (B.9) are symmetric, and each $A_{t,j}$ is symmetric. For $X, Y \in \mathbb{R}^{m\times n}$, write $x_j = X[:,j]$ and $y_j = Y[:,j]$. We obtain

$$\langle X, \mathcal{A}_t Y\rangle_F = \sum_{j=1}^{n} x_j^\top A_{t,j} y_j = \sum_{j=1}^{n} (A_{t,j} x_j)^\top y_j = \langle \mathcal{A}_t X, Y\rangle_F. \tag{B.10}$$

For the quadratic form of each column, expand the two operator components separately to get

$$\begin{aligned} x_j^\top A_{t,j} x_j &= \alpha_c (P_t^\top x_j)^\top \left(\widehat{H}_{t,j}^c + \epsilon I_r\right)^{-1} (P_t^\top x_j) + \frac{\gamma_t}{\sigma_{t,j}+\epsilon} x_j^\top \Pi_t^f x_j \\ &= \alpha_c \left\| \left(\widehat{H}_{t,j}^c + \epsilon I_r\right)^{-1/2} P_t^\top x_j \right\|_2^2 + \frac{\gamma_t}{\sigma_{t,j}+\epsilon} \|\Pi_t^f x_j\|_2^2. \end{aligned} \tag{B.11}$$

The second equality uses symmetry and idempotence of Π_t^f, so that $x_j^\top \Pi_t^f x_j = x_j^\top (\Pi_t^f)^\top \Pi_t^f x_j$. Summing Eq. (B.11) over j proves Eq. (B.6).

Every term is nonnegative when $\gamma_t \geq 0$. If $\gamma_t > 0$ and $x_j \neq 0$, the orthogonal decomposition gives

$$x_j = P_t P_t^\top x_j + \Pi_t^f x_j, \qquad \|x_j\|_2^2 = \|P_t^\top x_j\|_2^2 + \|\Pi_t^f x_j\|_2^2 > 0. \tag{B.12}$$

At least one term in Eq. (B.11) is therefore strictly positive. Every nonzero matrix has a nonzero column, which proves positive definiteness of $\mathcal{A}_t$. □

Lemma B.1 (Uniform spectral bounds). *Under Assumptions 5 and 6, the operator in Proposition B.1 satisfies $\mu I \preceq \mathcal{A}_t \preceq \nu I$, where*

$$\begin{aligned} \mu &= \min\left\{ \frac{\alpha_c}{G_{\max}+\epsilon}, \frac{\gamma_-}{G_{\max}/\sqrt{m-r}+\epsilon} \right\}, \\ \nu &= \frac{\max\{\alpha_c, \gamma_+\}}{\epsilon}. \end{aligned} \tag{B.13}$$

Proof. We first bound the gradients entering the two moment estimates. Since each column of P_k has unit norm and Π_k^f is an orthogonal projector, Assumption 5 gives, for $1 \le i \le r$ and $1 \le j \le n$,

$$\begin{aligned} \big|(P_k^\top G_k)[i,j]\big| &= \big|P_k[:,i]^\top G_k[:,j]\big| \\ &\le \|P_k[:,i]\|_2 \|G_k[:,j]\|_2 \le \|G_k\|_F \le G_{\max}, \\ \|\Pi_k^f G_k[:,j]\|_2 &\le \|G_k[:,j]\|_2 \le \|G_k\|_F \le G_{\max}. \end{aligned} \tag{B.14}$$

Starting from $V_0^c = 0$, repeated substitution in the compact second-moment recurrence yields

$$\begin{aligned} V_t^c[i,j] &= \beta_2 V_{t-1}^c[i,j] + (1-\beta_2)(P_t^\top G_t)[i,j]^2 \\ &= \beta_2^t V_0^c[i,j] + (1-\beta_2)\sum_{k=1}^{t} \beta_2^{t-k}(P_k^\top G_k)[i,j]^2 \\ &= (1-\beta_2)\sum_{k=1}^{t} \beta_2^{t-k}(P_k^\top G_k)[i,j]^2. \end{aligned} \tag{B.15}$$

All weights are nonnegative. Applying Eq. (B.14) and summing the geometric series gives

$$\begin{aligned} 0 \le V_t^c[i,j] &\le (1-\beta_2)G_{\max}^2 \sum_{k=1}^{t} \beta_2^{t-k} \\ &= (1-\beta_2)G_{\max}^2 \frac{1-\beta_2^t}{1-\beta_2} = (1-\beta_2^t)G_{\max}^2 \le G_{\max}^2. \end{aligned} \tag{B.16}$$

The bound remains valid when the basis changes because Eq. (B.14) holds separately for every historical basis P_k.

For the fine scale, let $e_{k,j} = \|\Pi_k^f G_k[:,j]\|_2^2/(m-r)$. Expanding the recurrence from $s_{0,j} = 0$ and then applying bias correction gives

$$\begin{aligned} s_{t,j} &= \beta_3 s_{t-1,j} + (1-\beta_3)e_{t,j} = (1-\beta_3)\sum_{k=1}^{t} \beta_3^{t-k} e_{k,j}, \\ \sigma_{t,j}^2 &= \widehat{s}_{t,j} = \frac{1-\beta_3}{1-\beta_3^t}\sum_{k=1}^{t} \beta_3^{t-k} e_{k,j}, \\ \frac{1-\beta_3}{1-\beta_3^t}\sum_{k=1}^{t} \beta_3^{t-k} &= \frac{1-\beta_3}{1-\beta_3^t}\frac{1-\beta_3^t}{1-\beta_3} = 1. \end{aligned} \tag{B.17}$$

Consequently, the corrected fine moment is a weighted average of energies bounded by $G_{\max}^2/(m-r)$, and

$$0 \le \sigma_{t,j}^2 \le \frac{1-\beta_3}{1-\beta_3^t}\sum_{k=1}^{t} \beta_3^{t-k}\frac{G_{\max}^2}{m-r} \le \frac{G_{\max}^2}{m-r}. \tag{B.18}$$

Taking square roots, adding ϵ, and inverting the positive denominators gives

$$\begin{aligned} \frac{\alpha_c}{G_{\max}+\epsilon} &\le \frac{\alpha_c}{\sqrt{V_t^c[i,j]}+\epsilon} \le \frac{\alpha_c}{\epsilon}, \\ \frac{\gamma_-}{G_{\max}/\sqrt{m-r}+\epsilon} &\le \frac{\gamma_t}{\sigma_{t,j}+\epsilon} \le \frac{\gamma_+}{\epsilon}. \end{aligned} \tag{B.19}$$

The middle expressions in Eq. (B.19) are the eigenvalues of the two diagonal blocks in Eq. (B.9). For any $X \in \mathbb{R}^{m\times n}$, write $x_j = X[:,j]$. Equations (B.11) and (B.19) then imply

$$\begin{aligned} \mu\big(\|P_t^\top x_j\|_2^2 + \|\Pi_t^f x_j\|_2^2\big) &\le x_j^\top A_{t,j} x_j \\ &\le \nu\big(\|P_t^\top x_j\|_2^2 + \|\Pi_t^f x_j\|_2^2\big). \end{aligned} \tag{B.20}$$

Using Eq. (B.12) and summing over the columns therefore yields

$$\mu\|X\|_F^2 \le \langle X, \mathcal{A}_t X\rangle_F \le \nu\|X\|_F^2. \tag{B.21}$$

This is the claimed operator inequality, with $\mu > 0$ because α_c, γ_-, and ϵ are positive. □

B.3 Direction, Momentum, and Update Errors

Lemma B.2 (Direction decomposition and descent)**.** *Define the compact momentum error and the update error by*

$$\begin{aligned} E_t &= M_t^c - P_t^\top \mathrm{g}_t, \\ Z_t[:,j] &= \alpha_c P_t \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} E_t[:,j] + \frac{\gamma_t}{\sigma_{t,j}+\epsilon} \Pi_t^f \xi_t[:,j]. \end{aligned} \tag{B.22}$$

If $0 \le \gamma_t \le \gamma_+$*, then the actual update satisfies*

$$D_t = \mathcal{A}_t \mathrm{g}_t + Z_t, \qquad \|Z_t\|_F^2 \le \frac{\alpha_c^2}{\epsilon^2}\|E_t\|_F^2 + \frac{\gamma_+^2}{\epsilon^2}\|\xi_t\|_F^2. \tag{B.23}$$

If Assumption 1, the spectral bounds of Lemma B.1, and $0 < h_t \le (2L\nu)^{-1}$ *hold, then*

$$f(W_{t+1}) \le f(W_t) - \frac{h_t}{2}\langle \mathrm{g}_t, \mathcal{A}_t \mathrm{g}_t\rangle_F + \frac{h_t}{2}\langle Z_t, \mathcal{A}_t^{-1} Z_t\rangle_F. \tag{B.24}$$

Proof. We first expand the stabilized update in each column. Substituting $M_t^c = P_t^\top \mathrm{g}_t + E_t$ and $G_t = \mathrm{g}_t + \xi_t$ into Eq. (8) gives

$$\begin{aligned} D_t[:,j] &= \alpha_c P_t \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} M_t^c[:,j] + \frac{\gamma_t}{\sigma_{t,j}+\epsilon}\Pi_t^f G_t[:,j] \\ &= \alpha_c P_t \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} P_t^\top \mathrm{g}_t[:,j] + \frac{\gamma_t}{\sigma_{t,j}+\epsilon}\Pi_t^f \mathrm{g}_t[:,j] \\ &\quad + \alpha_c P_t \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} E_t[:,j] + \frac{\gamma_t}{\sigma_{t,j}+\epsilon}\Pi_t^f \xi_t[:,j] \\ &= A_{t,j}\mathrm{g}_t[:,j] + Z_t[:,j]. \end{aligned} \tag{B.25}$$

This proves $D_t = \mathcal{A}_t \mathrm{g}_t + Z_t$. Denote the coarse and fine summands in Z_t by Z_t^c and Z_t^f. Because $P_t^\top \Pi_t^f = 0$, their inner product is

$$\langle Z_t^c, Z_t^f\rangle_F = \sum_{j=1}^n \frac{\alpha_c \gamma_t}{\sigma_{t,j}+\epsilon} E_t[:,j]^\top \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} P_t^\top \Pi_t^f \xi_t[:,j] = 0. \tag{B.26}$$

The isometry $\|P_t z\|_2 = \|z\|_2$ for every $z \in \mathbb{R}^r$ and the lower bound ϵ on every denominator imply

$$\begin{aligned} \|Z_t^c\|_F^2 &= \alpha_c^2 \sum_{j=1}^n \big\| \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} E_t[:,j]\big\|_2^2 \le \frac{\alpha_c^2}{\epsilon^2}\|E_t\|_F^2, \\ \|Z_t^f\|_F^2 &= \sum_{j=1}^n \frac{\gamma_t^2}{(\sigma_{t,j}+\epsilon)^2}\|\Pi_t^f \xi_t[:,j]\|_2^2 \le \frac{\gamma_+^2}{\epsilon^2}\|\xi_t\|_F^2. \end{aligned} \tag{B.27}$$

Expanding the squared norm and using Eq. (B.26) now gives the bound without an additional factor of two,

$$\begin{aligned} \|Z_t\|_F^2 &= \|Z_t^c\|_F^2 + 2\langle Z_t^c, Z_t^f\rangle_F + \|Z_t^f\|_F^2 \\ &\le \frac{\alpha_c^2}{\epsilon^2}\|E_t\|_F^2 + \frac{\gamma_+^2}{\epsilon^2}\|\xi_t\|_F^2. \end{aligned} \tag{B.28}$$

To prove descent, integrate along the segment from W_t to $W_{t+1} = W_t - h_t D_t$. The fundamental theorem of calculus and Assumption 1 yield

$$\begin{aligned} &f(W_t - h_t D_t) - f(W_t) + h_t\langle \mathrm{g}_t, D_t\rangle_F \\ &\quad = -h_t \int_0^1 \langle \nabla f(W_t - s h_t D_t) - \nabla f(W_t), D_t\rangle_F \,\mathrm{d}s \\ &\quad \le h_t \int_0^1 L s h_t \|D_t\|_F^2 \,\mathrm{d}s = \frac{L h_t^2}{2}\|D_t\|_F^2. \end{aligned} \tag{B.29}$$

Rearranging this inequality gives

$$f(W_{t+1}) - f(W_t) \le -h_t \langle \mathrm{g}_t, D_t \rangle_F + \frac{L h_t^2}{2} \|D_t\|_F^2. \tag{B.30}$$

Since $\mathcal{A}_t$ is positive definite, its positive square root and inverse square root exist. Set $u = \mathcal{A}_t^{1/2} \mathrm{g}_t$ and $v = \mathcal{A}_t^{-1/2} Z_t$. Self-adjointness and the direction decomposition imply

$$\begin{aligned} D_t &= \mathcal{A}_t^{1/2}(u+v), \\ \langle \mathrm{g}_t, D_t \rangle_F &= \langle \mathcal{A}_t^{-1/2} u, \mathcal{A}_t^{1/2}(u+v) \rangle_F = \|u\|_F^2 + \langle u, v \rangle_F, \\ \|D_t\|_F^2 &= \langle u+v, \mathcal{A}_t(u+v) \rangle_F \le \nu \|u+v\|_F^2. \end{aligned} \tag{B.31}$$

The last inequality uses the upper spectral bound. Inserting these expressions into Eq. (B.30) and using $L h_t \nu \le 1/2$ yields

$$\begin{aligned} f(W_{t+1}) - f(W_t) &\le -h_t(\|u\|_F^2 + \langle u, v \rangle_F) + \frac{L h_t^2 \nu}{2} \|u+v\|_F^2 \\ &\le -h_t(\|u\|_F^2 + \langle u, v \rangle_F) + \frac{h_t}{4} \|u+v\|_F^2. \end{aligned} \tag{B.32}$$

Cauchy Schwarz and Young's inequality give

$$-\langle u, v \rangle_F \le |\langle u, v \rangle_F| \le \|u\|_F \|v\|_F \le \frac{1}{2}\|u\|_F^2 + \frac{1}{2}\|v\|_F^2. \tag{B.33}$$

Expanding the square in Eq. (B.32) and then applying Eq. (B.33) gives

$$\begin{aligned} &- h_t(\|u\|_F^2 + \langle u, v \rangle_F) + \frac{h_t}{4} \|u+v\|_F^2 \\ &\quad = -\frac{3h_t}{4}\|u\|_F^2 - \frac{h_t}{2}\langle u, v \rangle_F + \frac{h_t}{4}\|v\|_F^2 \\ &\quad \le -\frac{3h_t}{4}\|u\|_F^2 + \frac{h_t}{4}\big(\|u\|_F^2 + \|v\|_F^2\big) + \frac{h_t}{4}\|v\|_F^2 \\ &\quad = -\frac{h_t}{2}\|u\|_F^2 + \frac{h_t}{2}\|v\|_F^2. \end{aligned} \tag{B.34}$$

Finally, the definitions of u and v give

$$\|u\|_F^2 = \langle \mathrm{g}_t, \mathcal{A}_t \mathrm{g}_t \rangle_F, \qquad \|v\|_F^2 = \langle Z_t, \mathcal{A}_t^{-1} Z_t \rangle_F. \tag{B.35}$$

Substitution into Eq. (B.34) proves Eq. (B.24). Every step holds for each realization, so the argument does not require the preconditioned noise to have zero conditional mean. □

Lemma B.3 (Accumulated compact momentum error)**.** *For $t \ge 2$, define*

$$q_t = P_{t-1}^\top \mathrm{g}_{t-1} - P_t^\top \mathrm{g}_t. \tag{B.36}$$

The error in Lemma B.2 obeys

$$\begin{aligned} E_1 &= -\beta_1 P_1^\top \mathrm{g}_1 + (1-\beta_1) P_1^\top \xi_1, \\ E_t &= \beta_1 E_{t-1} + \beta_1 q_t + (1-\beta_1) P_t^\top \xi_t \quad (t \ge 2). \end{aligned} \tag{B.37}$$

For every integer $T \ge 1$,

$$\sum_{t=1}^{T} \|E_t\|_F^2 \le \frac{3\beta_1^2}{1-\beta_1^2}\|\mathrm{g}_1\|_F^2 + \frac{3\beta_1^2}{(1-\beta_1)^2}\sum_{t=2}^{T}\|q_t\|_F^2 + 3\sum_{t=1}^{T}\|\xi_t\|_F^2. \tag{B.38}$$

Under Assumption 1,

$$\|q_t\|_F^2 \le 2L^2 h_{t-1}^2 \|D_{t-1}\|_F^2 + 2\|P_t - P_{t-1}\|_2^2 \|\mathrm{g}_{t-1}\|_F^2. \tag{B.39}$$

Proof. We derive the error recurrence directly from the compact momentum update,

$$M_t^c = \beta_1 M_{t-1}^c + (1-\beta_1) P_t^\top G_t, \qquad M_0^c = 0. \tag{B.40}$$

At the first iteration, the zero initial momentum gives

$$\begin{aligned} E_1 &= M_1^c - P_1^\top \mathrm{g}_1 \\ &= (1-\beta_1) P_1^\top (\mathrm{g}_1 + \xi_1) - P_1^\top \mathrm{g}_1 \\ &= -\beta_1 P_1^\top \mathrm{g}_1 + (1-\beta_1) P_1^\top \xi_1. \end{aligned} \tag{B.41}$$

For $t \geq 2$, substitute $G_t = \mathrm{g}_t + \xi_t$, and then add and subtract $P_{t-1}^\top \mathrm{g}_{t-1}$ inside the momentum term. This gives

$$\begin{aligned} E_t &= \beta_1 M_{t-1}^c + (1-\beta_1) P_t^\top (\mathrm{g}_t + \xi_t) - P_t^\top \mathrm{g}_t \\ &= \beta_1 \big(M_{t-1}^c - P_t^\top \mathrm{g}_t\big) + (1-\beta_1) P_t^\top \xi_t \\ &= \beta_1 \big(M_{t-1}^c - P_{t-1}^\top \mathrm{g}_{t-1}\big) + \beta_1 \big(P_{t-1}^\top \mathrm{g}_{t-1} - P_t^\top \mathrm{g}_t\big) \\ &\quad + (1-\beta_1) P_t^\top \xi_t \\ &= \beta_1 E_{t-1} + \beta_1 q_t + (1-\beta_1) P_t^\top \xi_t. \end{aligned} \tag{B.42}$$

This proves Eq. (B.37). To unfold the recurrence, first write the next two errors explicitly,

$$\begin{aligned} E_2 &= \beta_1 E_1 + \beta_1 q_2 + (1-\beta_1) P_2^\top \xi_2, \\ E_3 &= \beta_1^2 E_1 + \beta_1^2 q_2 + \beta_1 q_3 + (1-\beta_1)\big(\beta_1 P_2^\top \xi_2 + P_3^\top \xi_3\big). \end{aligned} \tag{B.43}$$

Continuing by induction gives, for every $t \geq 1$,

$$E_t = \beta_1^{t-1} E_1 + \beta_1 \sum_{k=2}^{t} \beta_1^{t-k} q_k + (1-\beta_1) \sum_{k=2}^{t} \beta_1^{t-k} P_k^\top \xi_k. \tag{B.44}$$

Indeed, multiplication of this identity by β_1 advances every historical exponent by one, and Eq. (B.42) supplies the two new terms with index $t+1$. Inserting Eq. (B.41) into the first term gives

$$\beta_1^{t-1} E_1 = -\beta_1^t P_1^\top \mathrm{g}_1 + (1-\beta_1) \beta_1^{t-1} P_1^\top \xi_1. \tag{B.45}$$

Combining its noise term with the last sum in Eq. (B.44) yields

$$E_t = -\beta_1^t P_1^\top \mathrm{g}_1 + \beta_1 \sum_{k=2}^{t} \beta_1^{t-k} q_k + (1-\beta_1) \sum_{k=1}^{t} \beta_1^{t-k} P_k^\top \xi_k. \tag{B.46}$$

The middle sum is empty at $t = 1$, whereas the last sum includes the current noise. For $\beta_1 = 0$, the recurrence gives $E_t = P_t^\top \xi_t$, consistent with this expansion when $\beta_1^0 = 1$.

We next bound the accumulated contribution of each term. For any matrix sequence x_k, the triangle inequality followed by weighted Cauchy Schwarz gives

$$\begin{aligned} \left\| \sum_{k=1}^{t} \beta_1^{t-k} x_k \right\|_F^2 &\leq \left(\sum_{k=1}^{t} \beta_1^{t-k} \|x_k\|_F \right)^2 \\ &= \left(\sum_{k=1}^{t} \sqrt{\beta_1^{t-k}} \sqrt{\beta_1^{t-k}} \|x_k\|_F \right)^2 \\ &\leq \left(\sum_{k=1}^{t} \beta_1^{t-k} \right) \left(\sum_{k=1}^{t} \beta_1^{t-k} \|x_k\|_F^2 \right) \\ &= \frac{1-\beta_1^t}{1-\beta_1} \sum_{k=1}^{t} \beta_1^{t-k} \|x_k\|_F^2 \\ &\leq \frac{1}{1-\beta_1} \sum_{k=1}^{t} \beta_1^{t-k} \|x_k\|_F^2. \end{aligned} \tag{B.47}$$

Summing over t, exchanging the order of the finite sums, and evaluating the remaining geometric series yields

$$\begin{aligned}\sum_{t=1}^{T}\left\|\sum_{k=1}^{t}\beta_1^{t-k}x_k\right\|_F^2 &\le \frac{1}{1-\beta_1}\sum_{t=1}^{T}\sum_{k=1}^{t}\beta_1^{t-k}\|x_k\|_F^2\\ &= \frac{1}{1-\beta_1}\sum_{k=1}^{T}\|x_k\|_F^2\sum_{t=k}^{T}\beta_1^{t-k}\\ &= \frac{1}{1-\beta_1}\sum_{k=1}^{T}\|x_k\|_F^2\frac{1-\beta_1^{T-k+1}}{1-\beta_1}\\ &\le \frac{1}{(1-\beta_1)^2}\sum_{k=1}^{T}\|x_k\|_F^2.\end{aligned} \tag{B.48}$$

For the initialization term in Eq. (B.46), orthogonal projection is contractive, so

$$\begin{aligned}\sum_{t=1}^{T}\|\beta_1^t P_1^\top \mathrm{g}_1\|_F^2 &= \left(\sum_{t=1}^{T}\beta_1^{2t}\right)\|P_1^\top \mathrm{g}_1\|_F^2\\ &= \frac{\beta_1^2(1-\beta_1^{2T})}{1-\beta_1^2}\|P_1^\top \mathrm{g}_1\|_F^2\\ &\le \frac{\beta_1^2}{1-\beta_1^2}\|\mathrm{g}_1\|_F^2.\end{aligned} \tag{B.49}$$

For the drift term, apply Eq. (B.48) with $x_1 = 0$ and $x_k = q_k$ for $k \ge 2$. This gives

$$\begin{aligned}\sum_{t=1}^{T}\left\|\beta_1\sum_{k=2}^{t}\beta_1^{t-k}q_k\right\|_F^2 &= \beta_1^2\sum_{t=1}^{T}\left\|\sum_{k=2}^{t}\beta_1^{t-k}q_k\right\|_F^2\\ &\le \frac{\beta_1^2}{(1-\beta_1)^2}\sum_{k=2}^{T}\|q_k\|_F^2.\end{aligned} \tag{B.50}$$

For the noise term, apply the same inequality with $x_k = P_k^\top \xi_k$. The factor $(1-\beta_1)^2$ cancels the convolution bound, yielding

$$\begin{aligned}\sum_{t=1}^{T}\left\|(1-\beta_1)\sum_{k=1}^{t}\beta_1^{t-k}P_k^\top\xi_k\right\|_F^2 &\le \frac{(1-\beta_1)^2}{(1-\beta_1)^2}\sum_{k=1}^{T}\|P_k^\top\xi_k\|_F^2\\ &\le \sum_{k=1}^{T}\|\xi_k\|_F^2.\end{aligned} \tag{B.51}$$

This bound uses contraction for each realization and remains valid when P_k depends on the current noise.

To combine the three contributions, apply the triangle inequality and Cauchy Schwarz to any three matrices a, b, c of the same dimensions,

$$\|a+b+c\|_F^2 \le \big(\|a\|_F + \|b\|_F + \|c\|_F\big)^2 \le 3\big(\|a\|_F^2 + \|b\|_F^2 + \|c\|_F^2\big). \tag{B.52}$$

Applying this inequality to Eq. (B.46) and then using the three accumulated bounds gives

$$\begin{aligned}\sum_{t=1}^{T}\|E_t\|_F^2 &\le 3\sum_{t=1}^{T}\|\beta_1^t P_1^\top \mathrm{g}_1\|_F^2 + 3\sum_{t=1}^{T}\left\|\beta_1\sum_{k=2}^{t}\beta_1^{t-k}q_k\right\|_F^2\\ &\quad + 3\sum_{t=1}^{T}\left\|(1-\beta_1)\sum_{k=1}^{t}\beta_1^{t-k}P_k^\top\xi_k\right\|_F^2\\ &\le \frac{3\beta_1^2}{1-\beta_1^2}\|\mathrm{g}_1\|_F^2 + \frac{3\beta_1^2}{(1-\beta_1)^2}\sum_{k=2}^{T}\|q_k\|_F^2 + 3\sum_{k=1}^{T}\|\xi_k\|_F^2.\end{aligned} \tag{B.53}$$

This proves Eq. (B.38).

It remains to bound q_t. Adding and subtracting $P_t^\top \mathrm{g}_{t-1}$ separates the change in the gradient from the change in the projection basis,

$$\begin{aligned} q_t &= P_{t-1}^\top \mathrm{g}_{t-1} - P_t^\top \mathrm{g}_{t-1} + P_t^\top \mathrm{g}_{t-1} - P_t^\top \mathrm{g}_t \\ &= P_t^\top(\mathrm{g}_{t-1} - \mathrm{g}_t) + (P_{t-1}^\top - P_t^\top)\mathrm{g}_{t-1}. \end{aligned} \tag{B.54}$$

For the first summand, projection contraction, smoothness, and the iterate update imply

$$\begin{aligned} \|P_t^\top(\mathrm{g}_{t-1} - \mathrm{g}_t)\|_F &\le \|\nabla f(W_{t-1}) - \nabla f(W_t)\|_F \\ &\le L\|W_t - W_{t-1}\|_F = L|h_{t-1}|\|D_{t-1}\|_F. \end{aligned} \tag{B.55}$$

For the second summand, the operator norm inequality gives

$$\|(P_{t-1}^\top - P_t^\top)\mathrm{g}_{t-1}\|_F \le \|P_t - P_{t-1}\|_2 \|\mathrm{g}_{t-1}\|_F. \tag{B.56}$$

Finally, apply $\|a+b\|_F^2 \le 2\|a\|_F^2 + 2\|b\|_F^2$ to Eq. (B.54) and insert these two estimates to obtain

$$\begin{aligned} \|q_t\|_F^2 &\le 2\|P_t^\top(\mathrm{g}_{t-1} - \mathrm{g}_t)\|_F^2 + 2\|(P_{t-1}^\top - P_t^\top)\mathrm{g}_{t-1}\|_F^2 \\ &\le 2L^2 h_{t-1}^2 \|D_{t-1}\|_F^2 + 2\|P_t - P_{t-1}\|_2^2 \|\mathrm{g}_{t-1}\|_F^2. \end{aligned} \tag{B.57}$$

This is Eq. (B.39). □

The basis term in Lemma B.3 measures variation in P_t. A sign change or a rotation within an unchanged subspace can change the meaning of the retained compact momentum coordinates.

Lemma B.4 (Explicit accumulated update error)**.** *Under Assumptions 1, 2, 3, 5, and $0 \le \gamma_t \le \gamma_+$ almost surely, define*

$$\begin{aligned} D_{\max}^2 &= \frac{(\alpha_c^2 + \gamma_+^2) G_{\max}^2}{\epsilon^2}, \\ C_{\mathrm{init}} &= \frac{3\alpha_c^2 \beta_1^2 G_{\max}^2}{\epsilon^2(1-\beta_1^2)}, \\ C_{\mathrm{drift}} &= \frac{6\alpha_c^2 \beta_1^2 L^2 D_{\max}^2}{\epsilon^2(1-\beta_1)^2}, \\ C_{\mathrm{basis}} &= \frac{6\alpha_c^2 \beta_1^2 G_{\max}^2}{\epsilon^2(1-\beta_1)^2}, \\ C_{\mathrm{noise}} &= \frac{3\alpha_c^2 + \gamma_+^2}{\epsilon^2}. \end{aligned} \tag{B.58}$$

For a deterministic effective step sequence h_t and every integer $T \ge 1$, the error in Lemma B.2 satisfies

$$\sum_{t=1}^{T} \mathbb{E}\|Z_t\|_F^2 \le C_{\mathrm{init}} + C_{\mathrm{drift}} \sum_{t=1}^{T-1} h_t^2 + C_{\mathrm{basis}} \mathcal{V}_T + \frac{C_{\mathrm{noise}} T \upsilon^2}{\mathsf{b}}. \tag{B.59}$$

Proof. We first establish deterministic bounds on the gradient, the compact momentum, and the update direction. By unbiasedness, conditional Jensen, and Assumption 5,

$$\begin{aligned} \|\mathrm{g}_t\|_F^2 &= \big\|\mathbb{E}[G_t \mid \mathcal{H}_{t-1}]\big\|_F^2 \\ &\le \mathbb{E}[\|G_t\|_F^2 \mid \mathcal{H}_{t-1}] \le G_{\max}^2. \end{aligned} \tag{B.60}$$

Unfolding Eq. (B.40) from $M_0^c = 0$ gives

$$M_t^c = (1-\beta_1) \sum_{k=1}^{t} \beta_1^{t-k} P_k^\top G_k. \tag{B.61}$$

Each $P_k^\top$ is contractive, so the triangle inequality and the geometric sum imply

$$\begin{aligned} \|M_t^c\|_F &\le (1-\beta_1) \sum_{k=1}^{t} \beta_1^{t-k} \|P_k^\top G_k\|_F \\ &\le (1-\beta_1) G_{\max} \sum_{k=1}^{t} \beta_1^{t-k} = (1-\beta_1^t) G_{\max} \le G_{\max}. \end{aligned} \tag{B.62}$$

The two update components in Eq. (B.25) satisfy

$$\begin{aligned}\|D_t^c\|_F^2 &= \alpha_c^2 \sum_{j=1}^{n} \big\| \big(\widehat{H}_{t,j}^c + \epsilon I_r\big)^{-1} M_t^c[:,j] \big\|_2^2 \le \frac{\alpha_c^2}{\epsilon^2} \|M_t^c\|_F^2, \\ \|D_t^f\|_F^2 &= \sum_{j=1}^{n} \frac{\gamma_t^2}{(\sigma_{t,j}+\epsilon)^2} \|\Pi_t^f G_t[:,j]\|_2^2 \le \frac{\gamma_+^2}{\epsilon^2} \|G_t\|_F^2.\end{aligned} \tag{B.63}$$

Since $P_t^\top \Pi_t^f = 0$, the coarse and fine updates are orthogonal. Inserting their bounds and Eq. (B.62) yields

$$\begin{aligned}\|D_t\|_F^2 &= \|D_t^c\|_F^2 + \|D_t^f\|_F^2 \\ &\le \frac{\alpha_c^2}{\epsilon^2} \|M_t^c\|_F^2 + \frac{\gamma_+^2}{\epsilon^2} \|G_t\|_F^2 \\ &\le \frac{(\alpha_c^2 + \gamma_+^2) G_{\max}^2}{\epsilon^2} = D_{\max}^2.\end{aligned} \tag{B.64}$$

We next control the accumulated drift. Taking expectations in Eq. (B.39) and using the deterministic step sequence gives

$$\begin{aligned}\mathbb{E}\|q_t\|_F^2 &\le 2L^2 h_{t-1}^2 \mathbb{E}\|D_{t-1}\|_F^2 + 2\mathbb{E}\big[\|P_t - P_{t-1}\|_2^2 \|\mathsf{g}_{t-1}\|_F^2\big] \\ &\le 2L^2 h_{t-1}^2 D_{\max}^2 + 2G_{\max}^2 \mathbb{E}\|P_t - P_{t-1}\|_2^2.\end{aligned} \tag{B.65}$$

The second line uses the almost-sure bound on g_{t-1} and does not require independence between the basis and the gradient. Sum from $t = 2$ to T and use $\mathcal{V}_T = \sum_{t=2}^{T} \mathbb{E}\|P_t - P_{t-1}\|_2^2$ to obtain

$$\begin{aligned}\sum_{t=2}^{T} \mathbb{E}\|q_t\|_F^2 &\le 2L^2 D_{\max}^2 \sum_{t=2}^{T} h_{t-1}^2 + 2G_{\max}^2 \sum_{t=2}^{T} \mathbb{E}\|P_t - P_{t-1}\|_2^2 \\ &= 2L^2 D_{\max}^2 \sum_{t=1}^{T-1} h_t^2 + 2G_{\max}^2 \mathcal{V}_T.\end{aligned} \tag{B.66}$$

For the noise contribution, the tower property and Assumption 3 give

$$\begin{aligned}\mathbb{E}\|\xi_t\|_F^2 &= \mathbb{E}\big[\mathbb{E}[\|\xi_t\|_F^2 \mid \mathcal{H}_{t-1}]\big] \le \frac{\upsilon^2}{\mathsf{b}}, \\ \sum_{t=1}^{T} \mathbb{E}\|\xi_t\|_F^2 &\le \frac{T\upsilon^2}{\mathsf{b}}.\end{aligned} \tag{B.67}$$

Now take expectations in Eq. (B.38) and substitute Eqs. (B.60), (B.66), and (B.67). Keeping the four contributions explicit gives

$$\begin{aligned}\sum_{t=1}^{T} \mathbb{E}\|E_t\|_F^2 &\le \frac{3\beta_1^2 G_{\max}^2}{1-\beta_1^2} + \frac{3\beta_1^2}{(1-\beta_1)^2} \left(2L^2 D_{\max}^2 \sum_{t=1}^{T-1} h_t^2 + 2G_{\max}^2 \mathcal{V}_T \right) + \frac{3T\upsilon^2}{\mathsf{b}} \\ &= \frac{3\beta_1^2 G_{\max}^2}{1-\beta_1^2} + \frac{6\beta_1^2 L^2 D_{\max}^2}{(1-\beta_1)^2} \sum_{t=1}^{T-1} h_t^2 \\ &\quad + \frac{6\beta_1^2 G_{\max}^2}{(1-\beta_1)^2} \mathcal{V}_T + \frac{3T\upsilon^2}{\mathsf{b}}.\end{aligned} \tag{B.68}$$

Finally, sum the update-error inequality in Eq. (B.23) and take expectations,

$$\sum_{t=1}^{T} \mathbb{E}\|Z_t\|_F^2 \le \frac{\alpha_c^2}{\epsilon^2} \sum_{t=1}^{T} \mathbb{E}\|E_t\|_F^2 + \frac{\gamma_+^2}{\epsilon^2} \sum_{t=1}^{T} \mathbb{E}\|\xi_t\|_F^2. \tag{B.69}$$

Substituting Eqs. (B.68) and (B.67) into this inequality yields

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}\|Z_t\|_F^2 &\le \frac{3\alpha_c^2\beta_1^2G_{\max}^2}{\epsilon^2(1-\beta_1^2)}+\frac{6\alpha_c^2\beta_1^2L^2D_{\max}^2}{\epsilon^2(1-\beta_1)^2}\sum_{t=1}^{T-1}h_t^2\\ &\quad+\frac{6\alpha_c^2\beta_1^2G_{\max}^2}{\epsilon^2(1-\beta_1)^2}\mathcal{V}_T+\frac{3\alpha_c^2Tv^2}{\epsilon^2\mathsf{b}}+\frac{\gamma_+^2Tv^2}{\epsilon^2\mathsf{b}}\\ &= C_{\text{init}}+C_{\text{drift}}\sum_{t=1}^{T-1}h_t^2+C_{\text{basis}}\mathcal{V}_T+\frac{C_{\text{noise}}Tv^2}{\mathsf{b}}.\end{aligned}\tag{B.70}$$

The two noise terms combine into $C_{\text{noise}}=(3\alpha_c^2+\gamma_+^2)/\epsilon^2$, and the other three coefficients match Eq. (B.58). This proves Eq. (B.59). □

B.4 Proof of the Main Theorem

Lemma B.5 (Stationarity bound with update error)**.** *Suppose Assumptions 1 and 4, uniform bounds $\mu I \preceq \mathcal{A}_t \preceq \nu I$ with $\mu > 0$, and $0 < h_t \le (2L\nu)^{-1}$ hold. Then, for every integer $T \ge 1$, pathwise,*

$$\frac{\sum_{t=1}^{T}h_t\|\nabla f(W_t)\|_F^2}{\sum_{t=1}^{T}h_t}\le\frac{2\Delta}{\mu\sum_{t=1}^{T}h_t}+\frac{\sum_{t=1}^{T}h_t\|Z_t\|_F^2}{\mu^2\sum_{t=1}^{T}h_t}.\tag{B.71}$$

Proof. Rearrange Eq. (B.24) from Lemma B.2 to put the gradient term on the left,

$$\frac{h_t}{2}\langle \mathsf{g}_t,\mathcal{A}_t\mathsf{g}_t\rangle_F\le f(W_t)-f(W_{t+1})+\frac{h_t}{2}\langle Z_t,\mathcal{A}_t^{-1}Z_t\rangle_F.\tag{B.72}$$

Summing this inequality from $t=1$ to T gives

$$\begin{aligned}\frac{1}{2}\sum_{t=1}^{T}h_t\langle \mathsf{g}_t,\mathcal{A}_t\mathsf{g}_t\rangle_F &\le \sum_{t=1}^{T}\big(f(W_t)-f(W_{t+1})\big)\\ &\quad+\frac{1}{2}\sum_{t=1}^{T}h_t\langle Z_t,\mathcal{A}_t^{-1}Z_t\rangle_F.\end{aligned}\tag{B.73}$$

The objective differences telescope. Assumption 4 then yields

$$\begin{aligned}\sum_{t=1}^{T}\big(f(W_t)-f(W_{t+1})\big)&=f(W_1)-f(W_{T+1})\\ &\le f(W_1)-f_*=\Delta.\end{aligned}\tag{B.74}$$

Substitution into Eq. (B.73) and multiplication by two gives

$$\sum_{t=1}^{T}h_t\langle \mathsf{g}_t,\mathcal{A}_t\mathsf{g}_t\rangle_F\le 2\Delta+\sum_{t=1}^{T}h_t\langle Z_t,\mathcal{A}_t^{-1}Z_t\rangle_F.\tag{B.75}$$

The spectral bounds imply that every eigenvalue of $\mathcal{A}_t$ lies in $[\mu,\nu]$. The eigenvalues of its inverse are their reciprocals, so

$$\nu^{-1}I\preceq\mathcal{A}_t^{-1}\preceq\mu^{-1}I.\tag{B.76}$$

Applying these bounds to the two quadratic forms gives

$$\langle \mathsf{g}_t,\mathcal{A}_t\mathsf{g}_t\rangle_F\ge\mu\|\mathsf{g}_t\|_F^2,\qquad\langle Z_t,\mathcal{A}_t^{-1}Z_t\rangle_F\le\frac{1}{\mu}\|Z_t\|_F^2.\tag{B.77}$$

Use the lower bound on the left of Eq. (B.75) and the upper bound on its right to obtain

$$\mu\sum_{t=1}^{T}h_t\|\mathsf{g}_t\|_F^2\le 2\Delta+\frac{1}{\mu}\sum_{t=1}^{T}h_t\|Z_t\|_F^2.\tag{B.78}$$

Since $\sum_{t=1}^{T}h_t>0$ and $\mu>0$, divide by $\mu\sum_{t=1}^{T}h_t$ to obtain

$$\frac{\sum_{t=1}^{T}h_t\|\mathsf{g}_t\|_F^2}{\sum_{t=1}^{T}h_t}\le\frac{2\Delta}{\mu\sum_{t=1}^{T}h_t}+\frac{\sum_{t=1}^{T}h_t\|Z_t\|_F^2}{\mu^2\sum_{t=1}^{T}h_t}.\tag{B.79}$$

Recalling $\mathsf{g}_t=\nabla f(W_t)$ proves Eq. (B.71). □

Proof of Theorem 1. Lemma B.1 provides $\mu I \preceq \mathcal{A}_t \preceq \nu I$ with $\mu > 0$. The assumed step restriction therefore permits application of Lemma B.5. With $h_t = h$, the total step size is $\sum_{t=1}^{T} h_t = hT$, and Eq. (B.71) becomes

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\|\mathsf{g}_t\|_F^2 &\le \frac{2\Delta}{\mu hT} + \frac{1}{\mu^2 hT}\sum_{t=1}^{T} h\|Z_t\|_F^2 \\ &= \frac{2\Delta}{\mu hT} + \frac{1}{\mu^2 T}\sum_{t=1}^{T}\|Z_t\|_F^2.\end{aligned} \tag{B.80}$$

Taking expectations and using linearity for the finite sums yields

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(W_t)\|_F^2 \le \frac{2\Delta}{\mu hT} + \frac{1}{\mu^2 T}\sum_{t=1}^{T}\mathbb{E}\|Z_t\|_F^2. \tag{B.81}$$

The constant step sequence is deterministic, so Lemma B.4 applies. Since $\sum_{t=1}^{T-1} h_t^2 = (T-1)h^2$, its bound gives

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(W_t)\|_F^2 &\le \frac{2\Delta}{\mu hT} \\ &\quad + \frac{1}{\mu^2 T}\left(C_{\mathrm{init}} + C_{\mathrm{drift}}(T-1)h^2 + C_{\mathrm{basis}}\mathcal{V}_T + \frac{C_{\mathrm{noise}}T\upsilon^2}{\mathsf{b}}\right) \\ &= \frac{2\Delta}{\mu hT} + \frac{C_{\mathrm{init}}}{\mu^2 T} + \frac{C_{\mathrm{drift}}h^2}{\mu^2}\frac{T-1}{T} \\ &\quad + \frac{C_{\mathrm{basis}}\mathcal{V}_T}{\mu^2 T} + \frac{C_{\mathrm{noise}}\upsilon^2}{\mu^2\mathsf{b}}.\end{aligned} \tag{B.82}$$

The factor T in the accumulated noise cancels with the normalization. Finally, $0 \le (T-1)/T \le 1$ and $C_{\mathrm{drift}} \ge 0$ imply

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\|\nabla f(W_t)\|_F^2 &\le \frac{2\Delta}{\mu hT} + \frac{C_{\mathrm{init}}}{\mu^2 T} + \frac{C_{\mathrm{drift}}h^2}{\mu^2} \\ &\quad + \frac{C_{\mathrm{basis}}\mathcal{V}_T}{\mu^2 T} + \frac{C_{\mathrm{noise}}\upsilon^2}{\mu^2\mathsf{b}}.\end{aligned} \tag{B.83}$$

This is Eq. (9). All terms are finite under the stated bounds, including the basis variation because $\|P_t - P_{t-1}\|_2 \le \|P_t\|_2 + \|P_{t-1}\|_2 = 2$. □

Under the stated assumptions, Theorem 1 bounds the average squared gradient norm for the combined coarse and fine updates. Preconditioning both spaces makes the guarantee apply to the full gradient. The optimization and initialization terms decrease with the number of iterations, while the step size, projection-basis variation, and stochastic noise govern the remaining error.

C Experimental Configurations and Additional Results

C.1 LLaMA C4 Configuration

We use eight NVIDIA A100 GPUs for the LLaMA experiments. The 60M & 130M models use short sequences of 256 tokens, while the 350M model uses intermediate-length sequences of 1024 tokens. AdamW uses a learning rate of 0.001 and all other methods use 0.01 at each scale. GaLore, APOLLO, FiRA, and OrthVMS use projection ranks of 128, 256, and 256 for the 60M, 130M, and 350M models, respectively. APOLLO-Mini uses rank one.

For zero-shot evaluation, we use the validation splits for BoolQ, RTE, HellaSwag, WinoGrande, and PIQA and the test splits for OpenBookQA, ARC-Easy, ARC-Challenge, SciQ, and MathQA. We report `acc` for BoolQ, RTE, and WinoGrande, and character-length-normalized `acc_norm` for the other seven tasks. Average is computed from the ten unrounded task scores with equal weights. The memory values come from the separate persistent-storage measurements.

C.2 FineWeb-Edu GLA Configuration

We a single NVIDIA H200 GPU for the GLA experiments. Models of all sizes use use long sequences of 4096 tokens. GaLore, APOLLO, FiRA, and OrthVMS use projection ranks of 128, 256, 256, and 512 for the 60M, 130M, 350M, and 1B models, respectively. APOLLO-Mini uses rank one. Table 5 concludes the model configurations and training budgets.

Table 5: FineWeb-Edu GLA model configurations and reported training budgets.

Scale	Layers	Hidden size	Parameters	Training steps
60M	8	512	6×10^7	10,000
130M	12	768	1.3×10^8	20,000
350M	24	1024	3.7×10^8	60,000
1B	24	2048	1.4×10^9	100,000

For 60M, 130M and 350M, we select the lowest final-checkpoint validation PPL in each method's reported learning-rate sweep (Tables 6, 7, 8, and 9).

Table 6: GLA-60M final PPL after 10K steps. Columns denote learning rates. Bold marks the lowest PPL in each row.

Method	0.003	0.006	0.01	0.015	0.02
AdamW	50.07	**49.88**	55.52	52.04	58.54
APOLLO	140.61	**128.62**	131.05	133.74	136.49
APOLLO-Mini	**98.40**	137.35	139.37	136.47	125.68
FiRA	98.29	**86.56**	90.29	93.07	97.93
OrthVMS (Ours)	58.93	48.47	49.86	47.86	**47.53**

Table 7: GaLore GLA-60M sweep.

Method	0.0003	0.0005	0.001	0.0015	0.002
GaLore	248.83	185.38	126.52	114.61	**105.96**

Table 8: GLA-130M final PPL after 20K steps. Columns denote learning rates. Bold marks the lowest PPL in each row.

Method	0.003	0.006	0.01	0.015	0.02
AdamW	**31.74**	32.25	32.55	32.79	36.68
GaLore	39.85	**37.64**	38.84	38.79	40.66
APOLLO	**64.37**	64.49	70.08	70.38	76.76
APOLLO-Mini	**49.65**	52.63	50.10	50.63	52.72
FiRA	**47.41**	48.76	49.26	48.83	51.07
OrthVMS (Ours)	31.68	31.12	**30.09**	30.72	32.16

Table 9: GLA-350M final validation PPL after 60K steps on the common learning-rate grid. Bold marks the lowest PPL in each row.

Method	0.003	0.006	0.01	0.015	0.02
AdamW	20.92	**20.25**	20.97	21.43	47.55
GaLore	25.81	**25.17**	26.90	26.73	42.25
APOLLO	39.14	**38.78**	42.61	44.31	45.07
APOLLO-Mini	**22.27**	23.12	25.74	25.55	27.40
FiRA	**28.53**	28.85	31.39	33.55	43.37
OrthVMS (Ours)	20.90	**20.79**	21.09	21.49	47.66