CURVATURE-CONDITIONED MULTISCALE MOMENTUM WITH SPHERE CONSTRAINTS FOR LLM PRETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretraining accounts for a large fraction of the total computational cost in LLM training. However, noise-dominant gradients and the highly ill-conditioned loss landscape bring severe challenges. Although modern adaptive optimizers such as AdamW and Muon have achieved great success in large-scale pretraining, their reliance on gradient normalization offers limited mitigation of the ill-conditioned curvature. The progress along flat directions (eigen-directions of small eigenvalues), which dominates the final loss reduction, remains relatively slow. To enhance training dynamics along flat directions, we propose a curvature-conditioned multiscale momentum method with sphere constraints, delivering steady acceleration in LLM pretraining. This multiscale momentum, applied only along flat directions, pairs a slow-decay component for noise reduction with a fast-decay component for rapid curvature adaptation, harnessing their complementary strengths. Crucially, we employ a sphere-constraint technique to prevent parameter inflation and effective learning rate collapse that would otherwise arise from a naive combination. Extensive experiments show that our proposed method significantly accelerates Muon across diverse architectures (dense, MoE) and model sizes (0.12B–5.1B parameters). Theoretically, we verify the acceleration effect and provide insights into the design principles underlying the flat-direction multiscale momentum.

1 INTRODUCTION

The optimizer is one of the most critical factors for training efficiency and continued scaling in the pretraining stage of large language models (LLMs). Several inherent challenges complicate LLM pretraining. Since the total number of training tokens vastly exceeds the per-step batch size, *noise dominates the stochastic gradient*: its magnitude far exceeds the true gradient signal. Moreover, the loss landscape is *highly nonconvex and ill-conditioned*, characterized by numerous flat directions alongside a few sharp ones. Notably, *the training dynamics along the flat directions primarily drive the final loss reduction* (Wen et al., 2025; Song et al., 2025). Since the learning rate is constrained by eigenvalues in sharp directions and the noise level (Cohen et al., 2021; Andreyev & Beneventano, 2025), progress along these crucial flat directions remains slow, hampering overall training.

Adaptive optimizers such as AdamW (Loshchilov & Hutter, 2017) and Muon (Jordan et al., 2024) employ coordinate- or block-wise normalization of gradient statistics. They have demonstrated stability and favorable scaling behavior in industrial-scale LLM training. However, this normalization (whitening) only partially compensates for the ill-conditioned curvature, and progress along the flat directions leaves room for improvement. Recent works (Wang et al., 2024; 2025; Zhou et al., 2025; Zhu et al., 2026) attempt to address this by estimating flat directions and manually amplifying the learning rate along them. Yet this approach also **amplifies the stochastic noise along flat directions**, yielding limited improvement and potentially destabilizing training. This motivates the question:

How to accelerate training dynamics along flat directions by reducing gradient noise?

We address this with a momentum mechanism tailored to flat directions. Our contributions are summarized as follows:

- **Algorithm design.** We propose a sphere-constrained multiscale momentum method that combines fast- and slow-decay momentum components to enhance training dynamics along the flat directions. The multiscale momentum is *curvature-conditioned*: it activates in flat

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.

directions for robust noise reduction and deactivates in sharp directions to preserve stability. On its own, however, it induces norm inflation and effective learning rate collapse. To address this, we introduce *sphere constraints to weights with parallel transport on each momentum component*, which prevents norm inflation and ensures proper inheritance of momentum across varying tangent spaces. Together, they *unlock the potential of multiscale momentum* in LLM pretraining.

- **Empirical evaluation.** We evaluate our proposed method (denoted by appending the suffix "M" to the base optimizer, e.g., MuonM) across a wide range of LLM pretraining settings, covering both dense and MoE architectures, model sizes from 0.12B to 5.1B, and diverse learning rate schedules (cosine decay and WSD). The results show that MuonM consistently outperforms Muon with lower terminal loss. Notably, these loss gains persist under extended token budgets, suggesting promising scalability to longer training horizons.
- **Theoretical analysis.** Recent studies (Lin et al., 2024; Li et al., 2025a; Meterez et al., 2026b) show that appropriately designed linear regression models can serve as a useful proxy for LLM pretraining dynamics, capturing key trends in loss evolution. Within this framework, we analyze how slow-decay momentum injection along flat directions accelerates optimization. Our analysis reveals that it *enables faster forgetting of accumulated noise while improving the signal learning, thereby leading to lower terminal loss*. These findings support the design intuition behind the multiscale momentum for flat directions.

Notations Define the matrix sign function as $\mathrm{msign}(M) = UV^\top$, where $M = U\Sigma V^\top$ is its singular value decomposition. The operator $\mathsf{RowScale}_r(M)$ scales each row of M to have a Frobenius norm of r, and $\mathsf{Norm}_r(M) = rM/\|M\|_F$. We use the term *curvature* in an intuitive sense to refer to Hessian information (e.g., eigenvalue distributions). The symbol $\asymp$ denotes Θ to absorb absolute constants. See Appendix B.1 for the full notation list.

2 Preliminaries and Related Works

Landscape and training dynamics in deep learning. Empirical studies (Ghorbani et al., 2019; Yao et al., 2020; Zhang et al., 2024; Tang et al., 2025) show that deep learning loss landscapes are highly ill-conditioned. Specifically, the Hessian possesses a small proportion of large positive eigenvalues (the corresponding eigenspaces are referred to as the *sharp directions*) and a massive number of near-zero and negative eigenvalues (corresponding to the *flat directions*). Recent studies (Song et al., 2025; Cohen et al., 2025; Wen et al., 2025) show that this landscape structure drives the **fast–slow training dynamics**: oscillatory behavior along sharp directions without divergence, and slow but persistent progress along flat directions that ultimately dominates the final loss reduction. Thus, accelerating optimization along flat directions is critical, and this is precisely the focus of our work.

Efficient adaptive optimizer design for LLM pretraining. Adaptive optimizers have become indispensable in LLM pretraining. AdamW (Loshchilov & Hutter, 2017) has become the de facto choice, owing to its simple and effective coordinate-wise preconditioning. However, its diagonal preconditioner inherently limits its capacity to capture cross-parameter correlations within the full curvature. Muon (Jordan et al., 2024; Liu et al., 2025) instead applies matrix-level preconditioning to parameter blocks, exhibiting favorable convergence speed and scaling behavior. Recent methods, such as (Vyas et al., 2025; Pethick et al., 2025; Li et al., 2026b), further develop structured matrix preconditioners to better approximate curvature-related information. Complementary to this line of work on preconditioner structure, we focus on momentum design instead.

Improved momentum mechanism for adaptive optimizers. Heavy Ball momentum (Polyak, 1964) has been widely adopted in modern adaptive optimizers to facilitate training. Building upon it, Nesterov-type momentum (Xie et al., 2024; Yuan et al., 2025) offers additional steady performance improvements. More recently, multiscale momentum has been introduced to smooth and accelerate the dynamics of momentum methods. AggMo (Lucas et al., 2019) linearly combines multiple momentum buffers with different decay rates to dampen oscillations. AdEMAMix (Pagliardini et al., 2025) extends this by integrating multiscale momentum with Adam's preconditioner in training neural networks. GPA (Defazio et al., 2026) employs extra primal averaging to smooth the iterates of Nesterov momentum. SODA (Pethick et al., 2026) and EMA-Nesterov (Yau et al., 2026) introduce an

extra momentum via primal extrapolation and exponential moving average of look-ahead directions respectively to achieve acceleration in pretraining, while accommodating various base optimizers. Different from these approaches which are motivated primarily by deterministic optimization, our work focuses on its role in variance reduction under noise-dominated stochastic environments, as well as on distinct behaviors at different time scales across regions of varying curvature (sharp and flat). In addition, we provide a unified high-order ODE formulation that offers a continuous-time perspective for understanding these different algorithmic forms in Appendix E.

Controlling the effective learning rate: from weight decay to sphere constraints. Normalization layers in modern LLMs introduce redundant degrees of freedom, making the learning rate alone insufficient to fully characterize the parameter update speed. For instance, an RMSNorm layer of the form $W_1\big(\gamma \odot \mathrm{Norm}(W_2 x)\big)$ exhibits a radial redundancy structure $W_1\,\mathrm{diag}(\gamma)$ and norm-scale invariance with respect to W_2 (when W_2 is learnable). This naturally motivates treating radial and angular updates separately. Recent studies (Roburin et al., 2022; Kosson et al., 2024; Wen et al., 2026; Zhou et al., 2026) regard the angular update rate $\|\Delta W\|_F/\|W\|_F$ as the *effective learning rate*, which has been empirically shown to play a dominant role in shaping the loss trajectory (Li et al., 2025b; Xiao et al., 2026; Liu et al., 2026). The radial evolution determines the weight norm and, in turn, affects the effective learning rate. Weight decay can be interpreted as controlling the noise-induced inflation of the weight norm, thereby preventing the effective learning rate from decaying too rapidly. Reference (Defazio, 2025) uses corrected weight decay coefficients to stabilize the weight norm. To achieve finer control, some recent approaches (Xie et al., 2026; Wen et al., 2026; An et al., 2026; Hägele et al., 2026) replace weight decay with direct norm constraints to weights, and they have demonstrated promising advantages over weight decay in LLM pretraining.

3 Illustrative Examples Motivating Flat-Direction Multiscale Momentum

In this section, we illustrate how varying momentum decay rates affect optimization dynamics.

3.1 Momentum Accelerates Flat-Direction Optimization by Reducing Variance

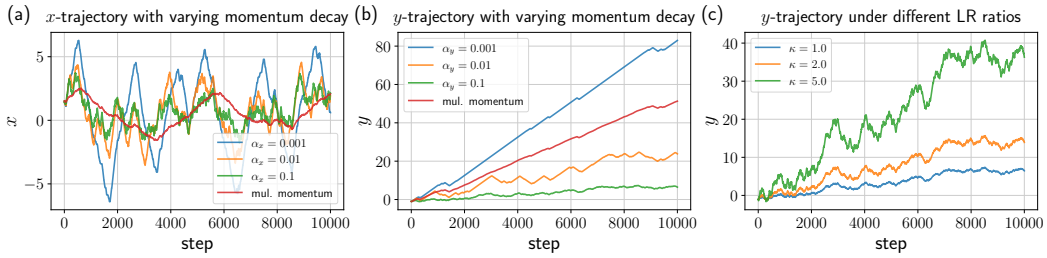

Figure 1: Coordinate-wise trajectories of different momentum methods for minimizing $f(x, y) = x^2 - 0.1y$ with the step size $\eta = 0.01$ and sign-momentum preconditioning. The x- and y-axes correspond to the sharp and flat directions, respectively. (a, b) employ various momentum decay coefficients α_x, α_y for x, y respectively. "mul. momentum" refers to (2) with $(\alpha^{\text{fast}}, \alpha^{\text{slow}}, c) = (0.1, 0.001, 0.75)$ and a biased flat-direction projection $\mathcal{Q}_k(x, y) = \frac{1}{\sqrt{0.2^2+0.8^2}}(0.2x + 0.8y)$. (c) amplifies the step size for y by a factor of κ.

We begin with a simple case: at iteration k, the stochastic gradient g_k follows $\mathcal{N}(\mu, \Sigma^2)$ with a constant true gradient μ, and the heavy-ball momentum updates as $m_k = (1-\alpha)m_{k-1} + \alpha g_k$. The coefficient $1-\alpha$ measures the momentum decay rate: **smaller α corresponds to slower decay, and larger α gives faster decay.** The estimator m_k has the same mean μ and a reduced equilibrium covariance $\frac{\alpha}{2-\alpha}\Sigma^2$. Thus, slower momentum decay yields lower-variance estimation of μ. However, the above reasoning requires the true gradient to be nearly unchanged. Since gradient variation is directly governed by its derivative (i.e., the Hessian), slow momentum decay is only safe in flat directions, where small Hessian eigenvalues lead to slowly changing gradients. Applying overly slow momentum decay to sharp directions can introduce bias and cause instability. A simple example

of minimizing $f(x, y) = x^2 - 0.1y$ with Gaussian gradient noise illustrates this clearly. Figure 1 (a,b) show that compared to a baseline with $\alpha = 0.1$, *using a smaller α (slower decay) for the flat direction y yields a faster and less noisy trajectory, whereas the same slow decay for the sharp direction x amplifies oscillations.* While increasing the learning rate for y also promotes progress, it simultaneously amplifies noise and leads to an oscillatory and more unstable dynamics (Figure 1 (c)). These observations motivate the use of anisotropic momentum decay to better suppress noise:

$$m_k = (I - \alpha_{\text{sharp}}\mathcal{P}_k - \alpha_{\text{flat}}\mathcal{Q}_k)m_{k-1} + g_k, \quad w_{k+1} = w_k - \eta_k m_k, \tag{1}$$

where $1 > \alpha_{\text{sharp}} \gg \alpha_{\text{flat}} > 0$ denote the decay coefficients for sharp and flat directions, and $\mathcal{P}_k$, $\mathcal{Q}_k$ denote projections onto the sharp and flat subspaces at the k-th step, respectively.

3.2 Flat-Direction Multiscale Momentum: Robust Variance Reduction against Projection Errors

In practice, the acceleration of (1) is compromised by various imperfections, including errors in estimating the flat-subspace projection and unmodeled variations in the Hessian eigenspace. Projection estimation errors introduce a portion of the slow decay for momentum in sharp directions, thereby destabilizing training. At the same time, persistent rotation and alternating transitions of the Hessian eigenspaces inject large bursts of sharp-direction momentum into the flat subspaces. Due to the slow forgetting in flat regions (requiring $\Theta(\alpha_{\text{flat}}^{-1})$ steps), such biases can persist and cause oscillations. To address this, we use multiscale momentum in flat directions, which linearly interpolates a fast-decay component m^{fast} for quick adaptation to gradient and curvature changes and a slow-decay component m^{slow} for variance reduction. At the same time, we retain the fast-decay momentum m^{fast} to help maintain stability. This leads to the following algorithm:

$$\begin{aligned} m_k^{\#} &= (1 - \alpha_{\#})m_{k-1}^{\#} + \alpha_{\#} g_k, \quad \# \in \{\text{fast}, \text{slow}\}, \\ w_{k+1} &= w_k - \eta_k\big(c\mathcal{Q}_k m_k^{\text{slow}} + (1-c)m_k^{\text{fast}}\big) \\ &= w_k - \eta_k\big(\mathcal{Q}_k(cm_k^{\text{slow}} + (1-c)m_k^{\text{fast}}) + (1-c)\mathcal{P}_k m_k^{\text{fast}}\big), \end{aligned} \tag{2}$$

where $\alpha_{\text{fast}} \gg \alpha_{\text{slow}} > 0$ denote the momentum decay rates, $0 < c < 1$ is the interpolation coefficient, and $\mathcal{P}_k$, $\mathcal{Q}_k$ are the projections onto sharp and flat subspaces. The interpolation $cm_k^{\text{slow}} + (1-c)m_k^{\text{fast}}$ also yields lower noise in estimating the true gradient compared to m_k^{fast} under the setting of Section 3.1 (Appendix F.1). As illustrated in Figure 1 (a, b) and Figure 2, with projection errors and varying curvature, the flat-direction multiscale momentum still yields *less noisy acceleration along the flat direction while not amplifying oscillations in the sharp direction.* The robust variance reduction of flat-direction multiscale momentum can be intuitively understood as a **bias–variance trade-off**. Please refer to Appendix G.2 for more details.

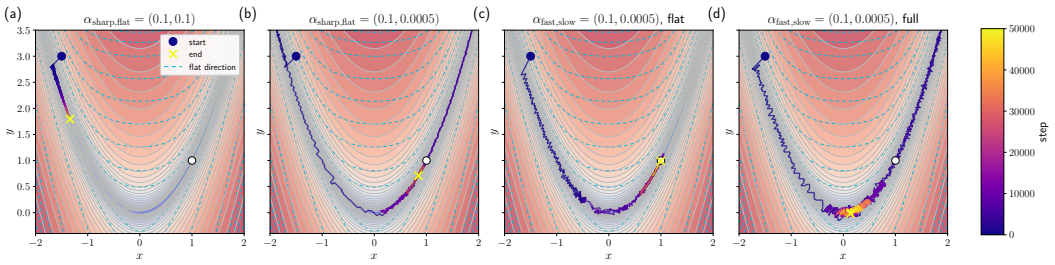

Figure 2: Optimization trajectories on $f(x, y) = (1 - x)^2 + 1000(y - x^2)^2$ (global minima at $(1, 1)$) under Gaussian noise. (a) Standard momentum progresses slowly. (b) Anisotropic momentum (1) advances faster along flat directions but induces back-and-forth oscillations. (c) Flat-direction multiscale momentum (2) ($c = 0.5$) achieves stable acceleration. (d) Total-space multiscale momentum ($c = 0.5$) leads to instability, implying that this technique should be restricted to flat directions.

4 Obstacles with Naive Multiscale Momentum: Parameter Inflation and Effective Learning Rate Collapse

We now extend the idea of (2) to the Muon optimizer for LLM pretraining. To estimate flat directions, we consider distinguishing them within each parameter block. Following (Zhu et al., 2026), we

empirically identify the bottom 90% singular spaces of the momentum matrix as flat directions, based on empirical evidence that the top singular spaces align well with the top eigenspaces of the Hessian. See implementation details in Appendix B.2. For a transformer matrix $W \in \mathbb{R}^{m\times n}$, we extend the preconditioner-free algorithm (2) by equipping m^{fast} with Muon preconditioning, while additionally incorporating the slow momentum m^{slow} in flat directions. The update rule [1] at the k-th iteration is

$$\begin{aligned} M_k^{\#} &= (1-\alpha_{\#})M_{k-1}^{\#} + G_k, \quad \# \in \{\text{fast}, \text{slow}\}, \\ W_{k+1} &= (1-\lambda\eta_k)W_k - \eta_k \left(\mathsf{msign}(\widetilde{M}_k^{\text{fast}}) + \chi\mathcal{Q}_k(\mathsf{RowScale}_{\min(1,\sqrt{\frac{n}{m}})}(M_k^{\text{slow}}))\right), \end{aligned} \tag{3}$$

where $\widetilde{M}_k^{\text{fast}} = (1-\alpha_{\text{fast}})M_k^{\text{fast}} + G_k$, λ is the weight decay coefficient, $\chi > 0$ is the slow-momentum coefficient, η_k is the scaled learning rate [2], and $\mathcal{Q}_k : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ is the estimated linear projection to flat directions. Rather than designing a specialized preconditioner for the slow momentum, we simply apply row-wise normalization to match the norm of the msign term. This approach is inspired by the near-zero inter-row Hessian observed in (Zhang et al., 2025), and it proves efficient in practice.

The results on the 0.12B dense model are in Figure 3 (left), where both the Muon baseline and the value of χ and α_{slow} have been thoroughly tuned (see details in Appendix C.1). The inclusion of slow momentum along flat directions yields a certain reduction in loss. However, the gain diminishes progressively, resulting in a terminal loss improvement of less than 0.01. This behavior is attributed to the amplified weight norm $\|W_k\|_F$ by slow momentum (e.g., the norm of an up projection block in Figure 3, right) and in turn accelerates the decay of the effective learning rate $\frac{\|\Delta W_k\|_F}{\|W_k\|_F}$. We provide an illustrative theoretical explanation to this phenomenon in Appendix F.2, which reveals its intrinsic link to the slow momentum. This mechanism also explains the evolution of the loss improvement: a sharp initial loss drop driven by the rapid effective learning rate decay in the early phase, followed by the baseline gradually catching up in the later phase, though the flat-direction slow-momentum acceleration still retains a marginal edge.

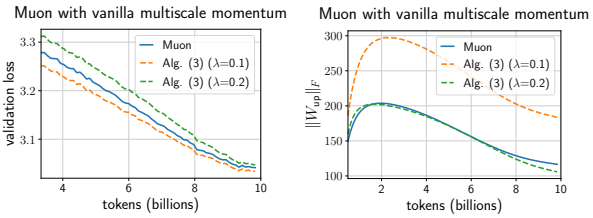

Figure 3: Straightforward application of flat-direction multiscale momentum to Muon on a 0.12B dense model. (Left) Validation loss; (Right) Frobenius norm evolution of an up projection block during training.

A natural remedy would be to use a larger weight decay coefficient λ (default 0.1) to restrain the norm inflation. However, even with $\lambda = 0.2$ which brings the norm back to the baseline level, the loss becomes even higher. This indicates that hyper-parameter tuning alone struggles to deliver notable acceleration for the naive combination in (3). We therefore turn to weight-decay-free techniques.

5 The Sphere-Constrained Flat-Direction Multiscale Momentum

We address the issue of weight norm inflation discussed in the previous section by replacing weight decay with **sphere constraints**, which necessitates manifold optimization techniques. A brief review of manifold optimization is provided in Appendix A, including key operations such as **tangent space projection** and **parallel transport** used in the proposed method below.

5.1 The Proposed Approach

Directional Sphere Parameterization with Learnable Radius To decouple radial and angular updates, we re-parameterize each transformer weight matrix W as $W = r\widehat{W}$, where $\|\widehat{W}\|_F \equiv R$ and $r \in \mathbb{R}$ is a learnable scalar. Motivated by (Wen et al., 2026), we rescale the optimizer update $\Delta\widehat{W}$ to the same norm R, which translates the learning rate into an angular velocity on the sphere. Compared to a purely fixed-norm parameterization, the learnable r automatically adapts to the natural

[1] Here we set the coefficient of G_k to 1. This choice differs from the EMA format in (2) only by a constant factor and is thus essentially equivalent.

[2] We use the RMS norm alignment rescaling and Nesterov-type momentum $\widetilde{M}^{\text{fast}}$ in (Liu et al., 2025).

heterogeneity across blocks and layers (Wang et al., 2026c). Vector parameters (e.g., the gain vectors in RMSNorm layers) as well as the embedding and output matrices are reparameterized analogously (see Appendix B.4 for more details). We denote a base optimizer used on the re-parameterized model by appending the suffix "S" (sphere) to its name (e.g., **MuonS**).

Algorithm Design We now apply the manifold optimization paradigm (parallel transport $\mathcal{T}$ in (9) and tangent space projection $\mathcal{P}$) from Section A to the base multiscale momentum method (3) under the sphere constraint described above. The resulting algorithm (denoted as **MuonM**) for transformer blocks is summarized in Algorithm 1, which degenerates to MuonS when $\chi = 0$. We rescale both the fast- and slow-momentum update norms to the sphere radius R (for the slow momentum, via the scaling factor $R/\sqrt{m}$), so that the learning rate acts as the angular velocity [3]. Nesterov-type momentum $\widetilde{M}_k^{\text{fast}} = (1-\alpha_{\text{fast}})M_k^{\text{fast}} + \mathcal{P}_{T_{W_k}\mathbb{S}_R}G_k$ is used as in (4). The flat-direction projection $\mathcal{Q}_k$ already includes the projection to the tangent space, and is computed by online power iterations to estimate top singular subspaces of $\widetilde{M}_k^{\text{fast}}$. Implementation details are provided in Section B.2. We apply Adam to scalar parameters and sphere-constrained Adam to vector parameters. Please refer to Section B.4 for the complete algorithm details. These additional operations incur negligible training-time overhead relative to Muon in practice (Appendix B.3).

Algorithm 1 MuonM for a transformer block matrix $W \in \mathbb{R}^{m\times n}$

1: **Input:** $\{\eta_k\}, \alpha_{\text{fast}} > \alpha_{\text{slow}} > 0, \chi \geq 0$.
2: **for** $k = 0$ **to** K **do**
3: Update momenta $M_k^{\#} = (1-\alpha_{\#})\mathcal{T}_{W_{k-1}\to W_k}(M_{k-1}^{\#}) + \mathcal{P}_{T_{W_k}\mathbb{S}_R}G_k, \quad \# \in \{\text{fast}, \text{slow}\}$.
4: Estimate projection $\mathcal{Q}_k$ onto the flat directions. Then compute the update direction

$$U_k = \mathsf{Norm}_R(\mathcal{P}_{T_{W_k}\mathbb{S}_R}\mathsf{msign}(\widetilde{M}_k^{\text{fast}})) + \chi\mathcal{Q}_k(\mathsf{RowScale}_{R/\sqrt{m}}(M_k^{\text{slow}})). \tag{4}$$

5: Update parameters $W_{k+1} = \mathsf{Norm}_R(W_k - \eta_k U_k)$.
6: **end for**

6 Experiments

In this section, we evaluate our proposed optimizer on dense and Mixture-of-Expert (MoE) architectures with scales from 0.12B to 5.1B. All models are trained on the high-quality FineWeb-Edu dataset (Lozhkov et al., 2024). The training budget is approximately 100 tokens per parameter (TPP) for dense models and approximately 500 tokens per activated parameter for MoE models, substantially exceeding the 20 TPP of the Chinchilla-optimal data (Hoffmann et al., 2022) and aligning more closely with industrial pretraining practice. Unless otherwise specified, the token batch size is about 0.5M for dense models and 4M for MoE models. More experimental details are in Appendices B, C.

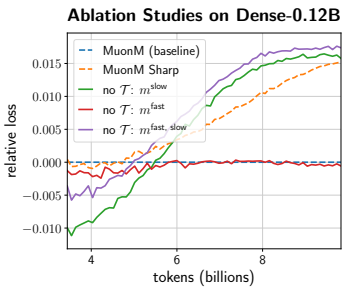

Figure 4: Ablation studies on projection direction and parallel transport $\mathcal{T}$. The label "Sharp" corresponds to replacing $\mathcal{Q}$ with the sharp-direction projection.

6.1 Validation of Critical Algorithm Components

We evaluate the roles of the *flat-direction projection* $\mathcal{Q}$ and *parallel transport* $\mathcal{T}$ for momentum on a 0.12B dense model, using MuonM as the baseline (Figure 4). We first replace the flat-direction projection $\mathcal{Q}$ with the sharp-direction projection (i.e., the orthogonal complement of $\mathcal{Q}$), which leads to a large increase in loss, indicating that **applying multiscale momentum along sharp directions is detrimental**. Next, we examine the effect of parallel transport on the fast and slow momentum separately. Removing parallel transport for the fast momentum alone results in a negligible change in the loss trajectory. In contrast, removing it for the slow momentum causes a significant increase in loss, and further disabling parallel transport for the

[3] Since $\|U_k\|_F$ remains stable throughout training (Figure 12), an extra re-normalization to R amounts to a nearly constant factor absorbable into learning rates and is therefore omitted.

fast momentum has little additional impact. This suggests that **parallel transport is unnecessary for the fast momentum but indispensable for the slow momentum**. A plausible explanation is that the fast momentum decays rapidly. Over its short decay horizon (e.g., half-life ≈ 14 steps at $1 - \alpha_{\text{fast}} = 0.95$), relatively few parameter updates occur and the tangent space changes little. Thus the accumulated error from skipping parallel transport remains small. The slow momentum, on the other hand, decays slowly (e.g., half-life ≈ 700 steps at $1 - \alpha_{\text{slow}} = 0.999$) and spans many updates over a large parameter range, during which the tangent space can change substantially. Parallel transport is therefore required to control the error accumulation.

6.2 Results on Dense and MoE Models

Comparison with Related Methods. We first compare our sphere constraint strategy (Section 5.1) with MuonH (Wen et al., 2026) (Frobenius norm sphere) and SSO (Xie et al., 2026) (spectral norm sphere). All sphere-constrained methods outperform the weight-decayed Muon baseline, with MuonS (i.e., MuonM with $\chi = 0$) achieving the lowest loss (Figure 5, left). We attribute this advantage to the adaptivity of the learnable radius across heterogeneous blocks. Next, we compare MuonM with other enhanced momentum approaches, including AdEMAMix (Pagliardini et al., 2025) (using AdamW as the base optimizer) and EMA-Nesterov (Yau et al., 2026) (using Muon as the base optimizer). MuonM achieves a lower terminal loss than these methods by a clear margin (Figure 5, right). Notably, the loss improvement of MuonM over MuonS (-0.019) and over Muon (-0.028) is substantially larger than the gain from directly applying flat-direction multiscale momentum to Muon (-0.008, see Figure 3). This indicates that **the sphere constraint more effectively unlocks the acceleration potential of flat-direction multiscale momentum.**

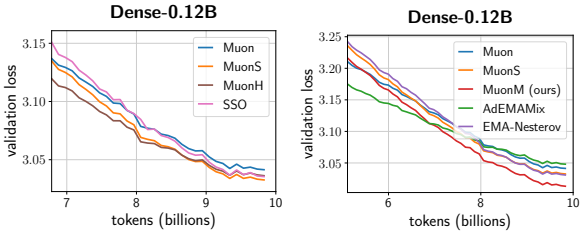

Figure 5: Comparison of MuonS and MuonM with other sphere-constrained methods (left) and momentum-enhancement approaches (right).

Main Results on Dense Models We systematically evaluate the acceleration benefits of MuonM against well-tuned MuonS and Muon baselines across dense models ranging from 0.12B to 1.4B parameters (Figures 6 and 5). Across all settings, MuonM **consistently achieves the lowest terminal loss**, outperforming MuonS and Muon by approximately 0.02 and 0.03, respectively.

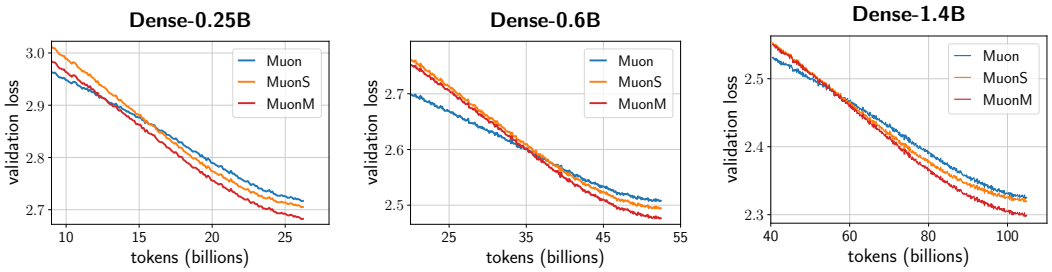

Figure 6: Validation losses on dense models across different optimizers. MuonM consistently achieves lower loss than all baselines with a notable margin.

Results on MoE Models Across MoE models from 0.6B to 5B parameters, MuonM consistently **achieves notable terminal loss reductions** over the well-tuned Muon and MuonS baselines (Figure 7). The acceleration also persists under the WSD schedule (Figure 13).

Scaling Behavior under Extended Training. We evaluate Muon, MuonS, and MuonM on the 0.12B dense model across token budgets ranging from approximately 100 to 1000 TPP, with a peak learning rate sweep performed at each budget. The terminal losses are fitted using

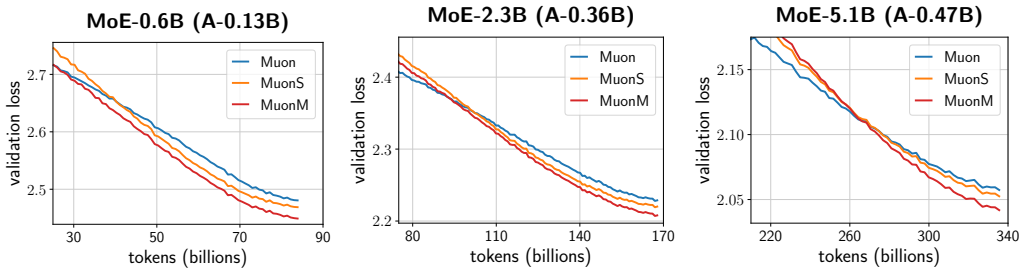

Figure 7: Validation losses on MoE models across different optimizers.

$L(N) = L_\infty + AN^{-\gamma}$ with the terminal loss $L(N)$, where N is the number of training tokens. As shown in Figure 8, MuonM consistently achieves the lowest terminal loss across all budgets, and the **performance gap widens as the training budget increases**, demonstrating its potential in long-horizon training. In addition, the searched optimal learning rate of MuonS (and thus MuonM) decays much more slowly (from at 1e-2 at 100 TPP to 7e-3 at 1000 TPP) compared to Muon, which decays from 5e-3 to 1e-3 over the same range (Table 3). **This mild shift in the optimal learning rate makes it more readily transferable across varying training horizons.**

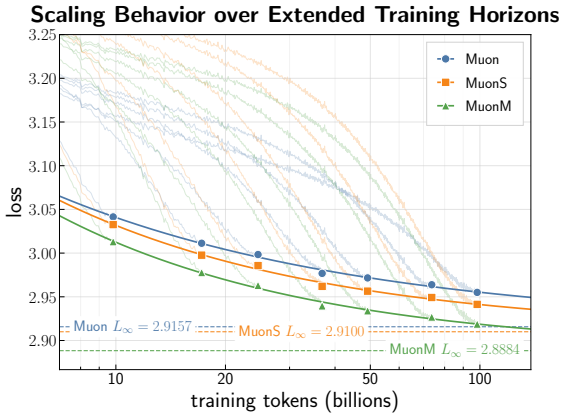

Figure 8: Validation losses on the 0.12B model under different token budgets. Dots indicate the terminal loss at each token budget with the corresponding tuned learning rate, and the solid curves are the fitted trends.

7 Theoretical Analysis

In this section, we analyze the training dynamics of flat-direction multiscale momentum and characterize its acceleration over vanilla momentum. Recent practices demonstrate the surprising effectiveness of quadratic models in approximating actual LLM training dynamics (Paquette et al., 2024; Lin et al., 2024; Li et al., 2025a; Meterez et al., 2026a;b). We adopt this setting to approximate the pretraining landscape. Specifically, we ground our analysis within the **Functional Scaling Law (FSL)** framework proposed by (Li et al., 2025a; 2026a), which accurately characterizes the entire loss dynamics beyond the traditional final-loss prediction.

7.1 Problem Formulation

Let $\mathcal{X}$ and $\mathcal{D}$ denote the data domain and data distribution, respectively. Let $\phi : \mathcal{X} \to \mathbb{R}^d$ be a feature map with $\phi(x) \sim \mathcal{N}(0, H)$ for $x \sim \mathcal{D}$, and let the target function be $f^\star(x) = \langle \phi(x), w^\star \rangle$ with target parameter $w^\star \in \mathbb{R}^d$. Assume $H = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ is diagonal with entries $\lambda_1 \geq \cdots \geq \lambda_d > 0$. Labels are generated according to $y(x) = \langle \phi(x), w^\star \rangle + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. We learn $f^\star$ using a student model $f(x, w) = \langle \phi(x), w \rangle$ with $w \in \mathbb{R}^d$ by minimizing the population risk

$$\mathcal{R}(w) = \frac{1}{2}\mathbb{E}_{x\sim\mathcal{D}}(f(x, w) - y(x))^2 = \frac{1}{2}u^\top H u + \frac{1}{2}\sigma^2, \tag{5}$$

where $u = w - w^\star$, and $\frac{1}{2}\sigma^2$ is the *irreducible loss*. For brevity, we define the excess risk $\mathcal{E}(w) = \frac{1}{2}u^\top H u$. We further assume that the problem exhibits the following power-law structure.

Assumption 7.1 (Power-law model)**.** There exist constants $\nu > 1$, $s > 0$, such that the Hessian spectrum and target parameter show a power-decayed structure with

$$\lambda_j \asymp j^{-\nu}, \quad |w_j^\star|^2 \asymp j^{-1}\lambda_j^{s-1} \asymp j^{-1-(s-1)\nu}, \quad 1 \leq j \leq d. \tag{6}$$

Let $\Sigma(w)$ denote the gradient noise covariance at w. The stochastic gradient at iteration k, computed on the mini-batch $\mathcal{B}$ at the parameter w_k, takes the form $g_k = \frac{1}{2|\mathcal{B}|}\sum_{x\in\mathcal{B}} \nabla_w (f(x, w_k) - y(x))^2 = Hu_k + \xi_k$, where the gradient noise satisfies $\mathbb{E}_k \xi_k = 0$ and has covariance $\mathbb{E}_k[\xi_k \xi_k^\top] = \Sigma(u_k)$. We consider the multiscale momentum with Nesterov-type momentum

$$m_k^{\#} = (1-\alpha_{\#})m_{k-1}^{\#} + g_k, \quad w_{k+1} = w_k - \eta_k\big(\chi Q m_k^{\text{slow}} + \gamma g_k + m_k^{\text{fast}}\big), \tag{7}$$

with $\# \in \{\text{fast}, \text{slow}\}$, initializations $w_0 = m_{-1}^{\text{fast}} = m_{-1}^{\text{slow}} = 0$ and hyper-parameters $0 < \alpha_{\text{slow}} < \alpha_{\text{fast}} \le 1, \chi \ge 0, \gamma \ge 0$. For an integer $0 \le r < d$, we denote the flat-direction projection as $Q = \sum_{j=1}^{d} q_j e_j e_j^\top$, where e_j is the eigenvector of λ_j and $q_j = \mathbb{I}_{\{j>r\}}$ is the indicator for flat directions. We define the intrinsic time as the accumulated learning rate $T_k := \sum_{i=0}^{k-1} \eta_i$.

7.2 Theoretical Results

Theorem 7.2 (FSL of Flat-Direction Multiscale Momentum)**.** *Define the constants* $a_{\text{top}} = \gamma + \alpha_{\text{fast}}^{-1}, a_{\text{full}} = \gamma + \alpha_{\text{fast}}^{-1} + \chi\alpha_{\text{slow}}^{-1}, p = 2 - \nu^{-1}$. *We further let* $a_j = a_{\text{top}}$ *for* $j \le r$ *and* $a_j = a_{\text{full}}$ *for* $j > r$. *Suppose Assumption 7.1 holds and* $d \ge 2(r+1)$. *Then for any* k *and learning rate schedule* $\{\eta_i\}$ *such that* $a_{\text{full}} T_k \lesssim d^\nu$ *and additional conditions in* (149) *hold, the iterates* w_k *in* (7) *satisfy*

$$\mathbb{E}\mathcal{E}(w_k) \asymp \mathsf{S}(T_k) + \sigma^2 \sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i), \tag{8}$$

where $\mathsf{S}(t) = \sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\text{top}}\lambda_j t} + \big((r+1)^\nu + a_{\text{full}} t\big)^{-s}$ *and* $\mathsf{K}(t) = \sum_{j=1}^{r} a_{\text{top}}^2 \lambda_j^2 e^{-2a_{\text{top}}\lambda_j t} + a_{\text{full}}^2\big((r+1)^\nu + a_{\text{full}} t\big)^{-p}$ *denote the signal learning and noise accumulation terms, respectively.*

The FSL in (8) provides a *close fit to the true loss trajectories* (Figure 9, further discussion in Appendix H). Using the same-order loss estimate from FSL, we show that flat-direction multiscale momentum *achieves lower loss* than the standard momentum method. We consider the constant learning rate setting for simplicity (see Corollary I.14 for more details).

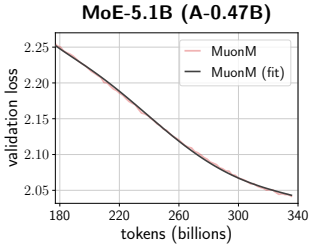

Figure 9: Comparison between the FSL-fitted loss and the true loss trajectory on the MoE-5.1B model.

Corollary 7.3 (Simplified Version of Corollary I.16)**.** *Suppose the conditions of Corollary I.14 hold. Define* $\kappa_\chi = \frac{\gamma + \alpha_{\text{fast}}{}^{-1} + \chi\alpha_{\text{slow}}{}^{-1}}{\gamma + \alpha_{\text{fast}}{}^{-1}}$ *and* $\Gamma_\chi = \frac{\sum_{j=1}^{r} j^{-\nu} + \sum_{j=r+1}^{d} j^{-\nu}}{\sum_{j=1}^{r} j^{-\nu}/\kappa_\chi + \sum_{j=r+1}^{d} j^{-\nu}} > 1$ *for* $\chi > 0$. *Let* $\mathcal{E}_\chi(\eta; k)$ *denote the asymptotic estimate of* $\mathbb{E}\mathcal{E}(w_k)$ *in* (8) *under a constant learning rate* η. *Then under the optimal learning rate* $\eta_{\chi,k}^\star = \operatorname{argmin}_\eta \mathcal{E}_\chi(\eta; k)$, *the loss reduction ratio is* $\frac{\mathcal{E}_\chi(\eta_{\chi,k}^\star; k)}{\mathcal{E}_0(\eta_{0,k}^\star; k)} \asymp \Gamma_\chi^{-\frac{s}{s+1}}$.

The corollary explicitly quantifies the speedup induced by slow momentum in flat directions, which becomes more pronounced for larger values of $\chi/\alpha_{\text{slow}}$.

8 Conclusion

In this work, we propose a flat-direction multiscale momentum method with sphere constraints for LLM pretraining. The sphere constraint serves as a more effective foundation than standard weight decay for unleashing the acceleration potential of flat-direction multiscale momentum, by resolving the effective learning rate collapse inherent to the multiscale design. Extensive experiments show that our proposed method (MuonM) consistently accelerates Muon with substantial loss reductions across diverse model architectures and training setups. Our results also indicate that the sphere-constraint technique holds promise as a general building block for developing more efficient optimizers.

Several limitations and open questions exist in this study. First, the optimal learning rate schedule for sphere-constrained methods remains an open question. Second, we have not investigated preconditioner design for the slow momentum, which we posit could further accelerate training.

AI USE STATEMENT

In this work, we used generative AI tools for polishing parts of the writing, for assisting with parts of the code, and for assisting with checking parts of the mathematical proofs. Generative AI tools were not used to develop or propose the research methodology or to evaluate or interpret the experimental results. Specifically, all AI-assisted text was manually reviewed and revised by the authors; all AI-assisted code was tested and verified for correctness by the authors. We take responsibility for the final content of this work, including the contents produced with the aid of generative AI.

ETHICS STATEMENT

The authors do not foresee any ethical concerns raised by this work.

REPRODUCIBILITY STATEMENT

All experiments in this work are conducted on publicly available datasets. The paper provides detailed descriptions of the proposed algorithm, including the full procedure, hyperparameter settings, and implementation details. The code used for our experiments is provided in the supplementary material to facilitate reproduction.

REFERENCES

Noah Amsel, David Persson, Christopher Musco, and Robert M. Gower. The polar express: Optimal matrix sign methods and their application to the muon algorithm. *arXiv preprint arXiv:2505.16932*, 2025.

Kang An, Jiaxiang Li, Donald Goldfarb, and Shiqian Ma. Demystifying manifold constraints in llm pre-training. *arXiv preprint arXiv:2605.04418*, 2026.

Arseniy Andreyev and Pierfrancesco Beneventano. Edge of stochastic stability: Revisiting the edge of stability for sgd. *arXiv preprint arXiv:2412.20553*, 2025.

Arseniy Andreyev, Advikar Ananthkumar, Marc Walden, Tomaso Poggio, and Pierfrancesco Beneventano. Momentum further constrains sharpness at the edge of stochastic stability. In *Forty-third International Conference on Machine Learning*, 2026.

Jeremy Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021.

Jeremy Cohen, Alex Damian, Ameet Talwalkar, J Zico Kolter, and Jason D. Lee. Understanding optimization in deep learning with central flows. In *The Thirteenth International Conference on Learning Representations*, 2025.

DeepSeek-AI, Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

Aaron Defazio. Why gradients rapidly increase near the end of training. *arXiv preprint arXiv:2506.02285*, 2025.

Aaron Defazio, Konstantin Mishchenko, Parameswaran Raman, Hao-Jun Michael Shi, and Lin Xiao. Smoothing diloco with primal averaging for faster training of llms. *arXiv preprint arXiv:2512.17131*, 2026.

Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via hessian eigenvalue density. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2232–2241. PMLR, 09–15 Jun 2019.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Alexander Hägele, Alejandro Hernández-Cano, Atli Kosson, and Martin Jaggi. Improving neural network training by decoupling the magnitude and direction of weight vectors. *arXiv preprint arXiv:2606.25971*, 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/`.

Atli Kosson, Bettina Messmer, and Martin Jaggi. Rotational equilibrium: How weight decay balances learning across neural networks. In *Forty-first International Conference on Machine Learning*, 2024.

Binghui Li, Fengling Chen, Zixun Huang, Lean Wang, and Lei Wu. Functional scaling laws in kernel regression: Loss dynamics and learning rate schedules. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a.

Binghui Li, Zilin Wang, Fengling Chen, Shiyang Zhao, Ruiheng Zheng, and Lei Wu. Optimal learning rate schedules under functional scaling laws: Power decay and warmup–stable–decay. In Steve Hanneke and Tor Lattimore (eds.), *Proceedings of Thirty Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 4722–4723. PMLR, 29 Jun–03 Jul 2026a.

Bingrui Li, Jiaxin Wen, Zhanpeng Zhou, Jun Zhu, and Jianfei Chen. Efficient hyperparameter tuning via trajectory invariance principle. *arXiv preprint arXiv:2509.25049*, 2025b.

Zichong Li, Liming Liu, Chen Liang, Weizhu Chen, and Tuo Zhao. Normuon: Making muon more efficient and scalable. In *Forty-third International Conference on Machine Learning*, 2026b.

Licong Lin, Jingfeng Wu, Sham M. Kakade, Peter Bartlett, and Jason D. Lee. Scaling laws in linear regression: Compute, parameters, and data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 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 preprint arXiv:2502.16982*, 2025.

Zihan Liu, Ruiheng Zheng, Shaobo Zhang, Changxin Tian, Kunlong Chen, Zhiqiang Zhang, and Lei Wu. Effective learning rate governs loss dynamics in language model pretraining. *arXiv preprint arXiv:2608.24814*, 2026.

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

Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. Fineweb-edu: the finest collection of educational content. *URL https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu*, 2024.

James Lucas, Shengyang Sun, Richard Zemel, and Roger Grosse. Aggregated momentum: Stability through passive damping. *arXiv preprint arXiv:1804.00325*, 2019.

Alexandru Meterez, Pranav Ajit Nair, Depen Morwani, Cengiz Pehlevan, and Sham Kakade. Anytime pretraining: Horizon-free learning-rate schedules with weight averaging. *arXiv preprint arXiv:2602.03702*, 2026a.

Alexandru Meterez, Pranav Ajit Nair, Depen Morwani, Cengiz Pehlevan, Sham Kakade, and Alex Damian. A defense of the quadratic model. *arXiv preprint arXiv:2607.21716*, 2026b.

Matteo Pagliardini, Pierre Ablin, and David Grangier. The adEMAMix optimizer: Better, faster, older. In *The Thirteenth International Conference on Learning Representations*, 2025.

Elliot Paquette, Courtney Paquette, Lechao Xiao, and Jeffrey Pennington. 4+3 phases of compute-optimal neural scaling laws. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained LMOs. In *Forty-second International Conference on Machine Learning*, 2025.

Thomas Pethick, Wanyun Xie, Roman Machacek, and Volkan Cevher. Optimistic dual averaging unifies modern optimizers. *arXiv preprint arXiv:2605.11172*, 2026.

B.T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964. ISSN 0041-5553.

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.

Simon Roburin, Yann de Mont-Marin, Andrei Bursuc, Renaud Marlet, Patrick Pérez, and Mathieu Aubry. Spherical perspective on learning with normalization layers. *arXiv preprint arXiv:2006.13382*, 2022.

Minhak Song, Kwangjun Ahn, and Chulhee Yun. Does SGD really happen in tiny subspaces? In *The Thirteenth International Conference on Learning Representations*, 2025.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Qian-Yuan Tang, Yufei Gu, Yunfeng Cai, Mingming Sun, Ping Li, Zhou Xun, and Zeke Xie. Investigating the overlooked hessian structure: From CNNs to LLMs. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 58805–58831. PMLR, 13–19 Jul 2025.

Nikhil Vyas, Depen Morwani, Rosie Zhao, Itai Shapira, David Brandfonbrener, Lucas Janson, and Sham M. Kakade. SOAP: Improving and stabilizing shampoo using adam for language modeling. In *The Thirteenth International Conference on Learning Representations*, 2025.

Jinbo Wang, Mingze Wang, Zhanpeng Zhou, Junchi Yan, Weinan E, and Lei Wu. The sharpness disparity principle in transformers for accelerating language model pre-training. In *Forty-second International Conference on Machine Learning*, 2025.

Jinbo Wang, Binghui Li, Zhanpeng Zhou, Mingze Wang, Yuxuan Sun, Jiaqi Zhang, Xunliang Cai, and Lei Wu. Fast catch-up, late switching: Optimal batch size scheduling via functional scaling laws. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Jinbo Wang, Mingze Wang, Jiaqi Zhang, Wei Wang, Peng Pei, Xunliang Cai, Weinan E, and Lei Wu. Gradpower: Powering gradients for faster language model pre-training. In *Forty-third International Conference on Machine Learning*, 2026b.

Mingze Wang, Jinbo Wang, Haotian He, Zilin Wang, Guanhua Huang, Feiyu Xiong, Zhiyu li, Weinan E, and Lei Wu. Improving generalization and convergence by enhancing implicit regularization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Mingze Wang, Shuchen Zhu, Yuxin Fang, Binghui Li, Kai Shen, and Shu Zhong. Negligible in size, significant in effect: On scale vectors in large language models. *arXiv preprint arXiv:2605.26895*, 2026c.

Kaiyue Wen, Zhiyuan Li, Jason S. Wang, David Leo Wright Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape view. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kaiyue Wen, Xingyu Dang, Kaifeng Lyu, Tengyu Ma, and Percy Liang. Fantastic pretraining optimizers and where to find them ii: Hyperball optimization. *arXiv preprint arXiv:2606.16899*, 2026.

Lei Wu, Mingze Wang, and Weijie J Su. The alignment property of SGD noise and how it helps select flat minima: A stability analysis. In *Advances in Neural Information Processing Systems*, 2022.

Yihao Xiao, Jialong Sun, Zitian Gao, Zeming Wei, Chutian Wang, Ran Tao, Jiaye Teng, and Bryan Dai. Hyperball may not be a free lunch. *arXiv preprint arXiv:2607.22444*, 2026.

Tian Xie, Haoming Luo, Haoyu Tang, Hu Yiwen, Jason Klein Liu, Qingnan Ren, Yang Wang, Xin Zhao, Rui Yan, Bing Su, Chong Luo, and Baining Guo. Controlled LLM training on spectral sphere. In *Forty-third International Conference on Machine Learning*, 2026.

Xingyu Xie, Pan Zhou, Huan Li, Zhouchen Lin, and Shuicheng Yan. Adan: Adaptive nesterov momentum algorithm for faster optimizing deep models. *arXiv preprint arXiv:2208.06677*, 2024.

Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W Mahoney. Pyhessian: Neural networks through the lens of the hessian. In *2020 IEEE international conference on big data (Big data)*, pp. 581–590. IEEE, 2020.

Chung-Yiu Yau, Dawei Li, Athanasios Glentis, Valentyn Boreiko, Hoi-To Wai, and Mingyi Hong. Ema-nesterov: Stabilizing nesterov's lookahead for accelerated deep learning optimization. *arXiv preprint arXiv:2605.25395*, 2026.

Huizhuo Yuan, Yifeng Liu, Shuang Wu, Zhou Xun, and Quanquan Gu. MARS: Unleashing the power of variance reduction for training large models. In *Forty-second International Conference on Machine Learning*, 2025.

Yushun Zhang, Congliang Chen, Tian Ding, Ziniu Li, Ruoyu Sun, and Zhi-Quan Luo. Why transformers need adam: A hessian perspective. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

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

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. Pytorch fsdp: Experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*, 2023.

Wenjie Zhou, Bohan Wang, Wei Chen, and Xueqi Cheng. Bsfa: Leveraging the subspace dichotomy to accelerate neural network training. *arXiv preprint arXiv:2510.25244*, 2025.

Zhanpeng Zhou, Wenjie Zhou, Yufei Gu, Juqiu Wang, Guanwei Zhang, Ruobing Xie, Xianbiao Qi, Yushi Bai, Kaifeng Lyu, and Shunyu Yao. From lr to elr: A better heuristic for pretraining dynamics, 2026. URL `https://hy.tencent.ai/research/elr`.

Shuchen Zhu, Rizhen Hu, Mingze Wang, Mou Sun, Xue Wang, Kun Yuan, and Zaiwen Wen. Accelerating llm pre-training through flat-direction dynamics enhancement. *arXiv preprint arXiv:2602.22681*, 2026.

Appendix

APPENDIX CONTENTS

A Preliminaries on Manifold Optimization **15**

B Algorithm Details **15**

B.1 Notations 15

B.2 Estimating Flat Directions 15

B.3 Computation, Communication and Memory Overhead Analysis for MuonM. 16

B.4 Update Rules of MuonM for Different Parameter Types 16

C Experimental Details **17**

C.1 Experimental Details of Section 4 17

C.2 Experimental Details of Section 6.2 18

D Additional Experimental Results **19**

E A Continuous-Time Perspective for Unifying Enhanced Momentum Methods **21**

E.1 Unified ODE Representation 24

F Illustrative Theoretical Analysis **24**

F.1 Variance Reduction of Multiscale Momentum 24

F.2 Weight Norm Inflation by Slow Momentum 25

G Discussions **26**

G.1 On the Complementary Design of Preconditioning and Momentum for Enhancing Training Dynamics along Flat Directions 26

G.2 Understanding Robust Variance Reduction of Flat-Direction Multiscale Momentum: A Bias-Variance Trade-off View 27

H Empirical Fitting of LLM Pre-training Loss Trajectory **27**

I Proof **29**

I.1 Proof of the Main Theorem 29

I.2 Acceleration of Flat-Direction Multiscale Momentum under Constant Learning Rates 49

I.3 Useful Lemmas 51

A PRELIMINARIES ON MANIFOLD OPTIMIZATION

Manifold optimization deals with optimization problems where the variable is constrained to lie on a manifold. In this work, we restrict ourselves to the Frobenius sphere constraint $\mathbb{S}_R = \{w \in \mathbb{R}^p : \|w\|_F = R\}$ with $R > 0$. The *tangent space* at a point $w \in \mathbb{S}_R$ is $T_w\mathbb{S}_R = \{v \in \mathbb{R}^p : v^\top w = 0\}$. To define the optimization geometry, the tangent space is equipped with a metric (a positive definite quadratic form). Here we simply use the Euclidean metric.

At the k-th step, the update direction Δw_k (e.g., gradient or momentum) is *projected onto the tangent space* via $\mathcal{P}_{T_{w_k}\mathbb{S}_R}(\Delta w_k) = \Delta w_k - \frac{\Delta w_k^\top w_k}{R^2} w_k$. A *retraction* $w_{k+1} = \mathsf{Norm}_R(w_k - \eta_k \Delta w_k)$ is then applied to keep the iterate on the sphere, where η_k is the step size. In this framework, momentum thus accumulates the Riemannian gradients $\mathcal{P}_{T_w\mathbb{S}_R}(g)$, with g denoting the Euclidean gradient. After moving from w_k to w_{k+1}, the momentum vector in $T_{w_k}\mathbb{S}_R$ is transferred to the new tangent space $T_{w_{k+1}}\mathbb{S}_R$. This is accomplished by *parallel transport* $\mathcal{T}_{w_k \to w_{k+1}} : T_{w_k}\mathbb{S}_R \to T_{w_{k+1}}\mathbb{S}_R$, which moves tangent vectors along the manifold while preserving their norms. Intuitively, parallel transport ensures the momentum is adjusted to the new geometry, preventing a gradual distortion of the accumulated information. For the Frobenius sphere, parallel transport admits a simple closed form. Omitting it would amplify direction errors in the accumulated gradient history, especially when momentum decays slowly and the tangent spaces vary substantially along the optimization trajectory. We will elaborate on this point in Section 6.1. Given $v \in T_w\mathbb{S}_R$ and a new point $w' \in \mathbb{S}_R$ ($w' \neq -w$), the transported vector is

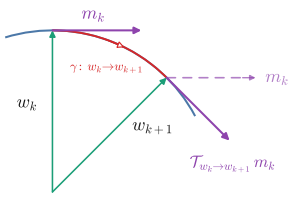

Figure 10: Illustration of parallel transport on the sphere.

$$\mathcal{T}_{w \to w'}(v) = v - \frac{\langle v, w' \rangle}{R^2 + \langle w, w' \rangle}(w + w'). \tag{9}$$

The above discussion can be directly extended to other tensor types such as matrices.

B ALGORITHM DETAILS

B.1 NOTATIONS

For a matrix $M \in \mathbb{R}^{m \times n}$, we define the matrix sign function as $\mathrm{msign}(M) = UV^\top$, where $M = U\Sigma V^\top$ is its singular value decomposition. Let $\|\cdot\|_F$ denote the Frobenius norm. The operator $\mathsf{RowScale}_r(M)$ scales each row of M to have a Frobenius norm of r, and $\mathsf{Norm}_r(M) = rM/\|M\|_F$. For a positive semi-definite matrix H, the induced semi-norm of a vector v is defined as $\|v\|_H = \sqrt{v^\top H v}$. The symbol $\odot$ represents the element-wise product. $\lesssim$, $\gtrsim$, and $\asymp$ denote $\mathcal{O}$, Ω, and Θ, respectively, to absorb absolute constants.

B.2 ESTIMATING FLAT DIRECTIONS

In this study, we focus on separating sharp and flat directions within each block. The construction of the flat-direction projection is motivated by the empirical observation in (Zhu et al., 2026) that, for a transformation block $W \in \mathbb{R}^{m \times n}$, the top eigenspaces of $G^\top G$ and $GG^\top$ are highly aligned with the per-row and per-column Hessians, respectively (Figure 11). Therefore, at the k-th iteration, we estimate the top singular space of the Nesterov-type momentum $\widetilde{M}_k^{\text{fast}} := (1-\alpha_{\text{fast}})M_k^{\text{fast}} + G_k$, which serves as a stabilized proxy for the gradient G_k, as the sharp directions. Specifically, we maintain a projection matrix $P^{\text{right}} \in \mathbb{R}^{n \times d}$ ($d \ll n$, e.g.,

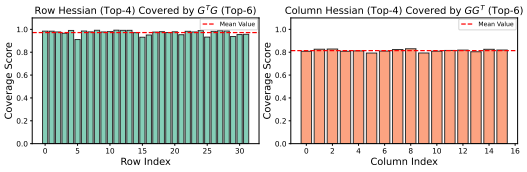

Figure 11: (Figure 3 in (Zhu et al., 2026)) Coverage of the top eigenspaces of row (column) Hessians by those of $G^\top G$ ($GG^\top$) for the `up_proj` block of an FFN layer in a toy dense language model, where $G \in \mathbb{R}^{32 \times 16}$. A higher score indicates a greater degree of containment.

$d = 0.1n$ in our experiments) that captures the
sharp directions of the row-wise Hessian. At iteration k, P^{right} is updated via online power iteration to track the top d right singular spaces:

$$P_k^{\text{right}} = \text{msign}\left(\left(\widetilde{M}_k^{\text{fast}}\right)^\top \widetilde{M}_k^{\text{fast}} P_{k-1}^{\text{right}}\right),$$

and the corresponding projection for the column-wise Hessian is then computed as

$$P_k^{\text{left}} = \text{msign}\big(\widetilde{M}_k^{\text{fast}} P_k^{\text{right}}\big) \in \mathbb{R}^{m\times d}.$$

Finally, for any $U \in \mathbb{R}^{m\times n}$, the flat-direction projection is given by

$$\mathcal{Q}_k(U) = \mathcal{P}_{T_{W_k}\mathbb{S}_R}\left(\left(I_m - P_k^{\text{left}}(P_k^{\text{left}})^\top\right) U \left(I_n - P_k^{\text{right}}(P_k^{\text{right}})^\top\right)\right),$$

where I denotes the identity matrix, $\mathcal{P}_{T_{W_k}\mathbb{S}_R}$ is the projection to the tangent space at W_k. Similarly, (Robert et al., 2025) applies the power method to estimate the top eigenspaces. Here we employ the more efficient msign operator to estimate the subspace projection, replacing the computationally expensive QR decomposition.

For the embedding and output matrices $W \in \mathbb{R}^{V\times d_{\text{model}}}$, where V and d_{model} denote the vocabulary size and model dimension, respectively, we simply select the d rows with the largest norm of the second moment in Adam. This choice is motivated by (Zhang et al., 2025), which shows that the token-wise Hessian exhibits an approximately block-diagonal structure across rows, each row (token) corresponding to a dense sub-block. In practice we set $d \ll V$ (e.g., $d = 0.1V$ in our experiments). For vector and scalar parameters, we do not apply flat-direction acceleration.

Computation and Memory Overhead of Flat-Direction Projection. The computational cost (in FLOPs) for computing $M \in \mathbb{R}^{m\times n}$ in computing $\mathsf{msign}(M)$ via Newton-Schulz (NS) iterations is approximately $6sr^2c$, where $c = \max\{m,n\}, r = \min\{m,n\}$, and s denotes the number of iterations. Consequently, compared to performing NS iterations on the full-dimensional matrix of size $m \times n$, employing the right and left projection matrices of sizes $m \times d$ and $n \times d$ reduces the computation to approximately $(d/n)^2$ of the original cost (e.g., $0.1 \times 0.1 = 0.01$) , rendering it nearly negligible. The additional memory overhead primarily stems from storing the right projection matrix of size $n \times d$, which is significantly smaller than that required for the momentum.

B.3 Computation, Communication and Memory Overhead Analysis for MuonM.

MuonM incurs only marginal additional overhead in computation, memory, and communication compared to the base Muon optimizer. The extra computation stems mainly from manifold-related operations (tangent space projection, retraction, and parallel transport) and the flat-direction projection $\mathcal{Q}_k$. The former involves no dense matrix–matrix products while the latter's main cost consists of two lightweight msign operations on matrices much smaller than W, which is nearly negligible as analyzed in Section B.2. The additional memory comes from the slow momentum buffer and the small projection matrices used by $\mathcal{Q}_k$, totaling roughly the size of one extra set of model parameters. In pretraining, memory is typically dominated by activations, and the extra optimizer state can be further reduced to a marginal amount per GPU via Fully Sharded Data Parallelism (FSDP) (Zhao et al., 2023). Regarding communication, in large-scale distributed training that employs parallelism strategies such as expert parallelism (EP) or pipeline parallelism (PP), the communication volume is similarly dominated by activations, and the extra communication from the sharded optimizer states is insignificant.

For instance, in the MoE-5.1B experiment (Figure 7) trained with the Megatron framework using EP=8, data parallelism, and a distributed optimizer that shards optimizer states, the average per-step iteration time is 2.8826 s for Muon and 2.9087 s for MuonM, an increase of only 0.9%.

B.4 Update Rules of MuonM for Different Parameter Types

Notations. For a matrix A, we use $A[i,:]$ to denote its i-th row. For a vector v, $v[i]$ denotes its i-th element.

We categorize the parameters into the following three types based on their structure:

1. **Matrix parameters**: Transformer weight matrices, such as Q, K, V, O, and FFN blocks. We apply MuonM (Algorithm 1) to these.
2. **Vector parameters**: biases (if any), learnable vectors in RMSNorm layers, each row of embedding/output layers (corresponding to each token), and the learnable grouped radii introduced by the sphere constraint. For these, we use Adam with sphere constraints, with slow momentum applied to partial vector blocks.
3. **Scalar parameters**: native architectural scalars and the learnable radii from the sphere constraint. We directly apply Adam to these.

The specific update rules are as follows.

- **Scalar.** It is initialized as 1 and trained directly by Adam without weight decay. The learning rate is the same as other vector and matrix parameters. In practice, if several scalars are interchangeable (e.g., the scalar multipliers of a weight matrix and its preceding RMSNorm, of the query and key matrices in attention, or of the up and down matrices in the FFN), we group them into a single learnable parameter.
- **Vector.** For a vector $\gamma \in \mathbb{R}^d$, we parameterize it as $\gamma = r\,\widehat{\gamma}$, where $r \in \mathbb{R}$ and $\widehat{\gamma} \in \mathbb{R}^d$ has a fixed norm R. At the k-th iteration, it is updated by

$$\begin{aligned} m_k &= (1-\alpha_{\text{fast}}^{\text{adam}})\,\mathcal{T}_{\widehat{\gamma}_{k-1}\to\widehat{\gamma}_k}(m_{k-1}) + \mathcal{P}_{T_{\widehat{\gamma}_k}\mathbb{S}_R}\,g_k, \\ v_k &= (1-\beta_{\text{adam}})\,v_{k-1} + \beta_{\text{adam}}\,g_k \odot g_k, \\ \widehat{\gamma}_{k+1} &= \mathsf{Norm}_R\Big(\widehat{\gamma}_k - \eta_k \mathcal{P}_{T_{\widehat{\gamma}_k}\mathbb{S}_R}\Big(\frac{m_k}{\sqrt{v_k}+\epsilon}\Big)\Big), \end{aligned} \tag{10}$$

  where g_k is the stochastic gradient.
- **Embedding and Output.** They are re-parameterized as $W = \mathrm{diag}(\gamma)\,\widehat{W}$, and we impose row-wise sphere constraints on $\widehat{W} \in \mathbb{R}^{V\times d}$, where the i-th row has a fixed Frobenius norm R_i. The vector γ is treated as a vector parameter and follows the vector update rules above. At the k-th iteration, for each row i we update the momenta as

$$M_k^{\#}[i,:] = (1-\alpha_{\#}^{\text{adam}})\,\mathcal{T}_{W_{k-1}[i,:]\to W_k[i,:]}\,(M_{k-1}^{\#}[i,:]) + \mathcal{P}_{T_{W_k[i,:]}\mathbb{S}_{R_i}}\,G_k[i,:], \tag{11}$$

  where G_k is the stochastic gradient, $\# \in \{\text{fast}, \text{slow}\}$, and parallel transport is performed row-wise. Then the flat-direction projection $\mathcal{Q}_k$ is computed as described in Section B.2, yielding a vector $q_k \in [0,1]^V$, where a value of 1 indicates that the corresponding row is selected as a flat direction. The update direction for each row is then

$$U_k[i,:] = \mathsf{Norm}_{R_i}\Big(\mathcal{P}_{T_{W_k[i,:]}\mathbb{S}_{R_i}}\Big(\frac{\widetilde{M}_k^{\text{fast}}[i,:]}{\sqrt{V_k[i,:]}+\epsilon}\Big)\Big) + \chi\,q_k[i]\,\mathsf{Norm}_{R_i}\big(M_k^{\text{slow}}[i,:]\big), \tag{12}$$

  where the second moment V_k is maintained as $V_k = (1-\beta_{\text{adam}})\,V_{k-1} + \beta_{\text{adam}}\,G_k \odot G_k$. Finally, the parameters are updated row-wise via

$$W_{k+1}[i,:] = \mathsf{Norm}_{R_i}\big(W_k[i,:] - \eta_k\,U_k[i,:]\big). \tag{13}$$

C Experimental Details

We use 6 Newton-Schulz (NS) iterations with coefficients in Polar Express (Amsel et al., 2025) to implement the msign operation, which achieves higher precision than the method in (Jordan et al., 2024). The peak learning rate is searched in {5e-4, 7e-4, 1e-3, 2e-3, 3e-3, 5e-3, 7e-3, 1e-2, 1.5e-2, 2e-2} for all optimizers.

C.1 Experimental Details of Section 4

In our hyperparameter search, the optimal learning rate for Muon was found to be 0.005. The sharp subspace ratio was set to 10%, following the procedure described in Appendix B.2. Beyond the learning rate, we set $\alpha_{\text{fast}} = 0.05$, searched over $\alpha_{\text{slow}} \in \{0.005, 0.002, 0.001, 0.0005\}$ and $\chi \in \{0.1, 0.2, 0.5\}$. The final best configuration is $\text{lr} = 0.005$, $\alpha_{\text{slow}} = 0.002$, and $\chi = 0.1$.

C.2 Experimental Details of Section 6.2

Comparing Different Optimizers on the 0.12B Dense Model. We use the cosine decay learning rate schedules for all optimizers. For MuonH and SSO, the optimal learning rates found by grid search are 1×10^{-2} and 5×10^{-3}, respectively. We also tuned the radius for SSO over $\{1.0, 2.0, 5.0\}$ and found that 2.0 yields the lowest loss. For AdEMAMix, we searched $\beta_3 \in \{0.01, 0.001, 0.0001\}$ and $\alpha \in \{1, 3, 5, 8\}$, obtaining the best configuration $(\mathrm{lr}, \beta_3, \alpha) = (3 \times 10^{-3}, 1 \times 10^{-3}, 5.0)$. Following (Pagliardini et al., 2025), the warmup iterations T_α and T_{β_3} are set to the total number of iterations. For EMA-Nesterov, we searched $\gamma \in \{0.99, 0.995, 0.999\}$ and $\beta_{\max} \in \{0.3, 0.5, 0.7, 0.9\}$, and selected $(\mathrm{lr}, \beta_{\max}, \gamma) = (3 \times 10^{-3}, 0.9, 0.99)$. The coefficient schedule of β_t involves two transition points T_w and T_r, and we adopted the recommended values $T_w = 0.3\,T$ and $T_r = 0.8\,T$ from (Yau et al., 2026), where T denotes the total number of iterations.

Muon Configuration. We follow the standard Muon setup described in (Liu et al., 2025). Muon is applied exclusively to all 2D transformer weight matrices, while AdamW handles all other parameters (RMSNorm, embeddings, and the output layer). For a Muon-updated matrix of shape $m \times n$, the learning rate is scaled by $0.2\sqrt{\max\{m, n\}}$ to align the RMS of the updates with that of AdamW. We employ Nesterov momentum with decay rate $\theta_{\text{muon}} = 0.95$, weight decay $\lambda = 0.1$, and a gradient clipping threshold of 1.0.

MuonM Configuration. Unless otherwise specified, we uniformly adopt $\chi = 0.2$, $\alpha_{\text{fast}} = 0.05$, $\alpha_{\text{slow}} = 0.001$, and set the top 10% subspace as sharp directions for each block using MuonM. Given the rapid gradient variations during the early training phase, we apply a warmup mechanism to the slow momentum coefficient α_{slow} as

$$\alpha_{\text{slow},t} = \begin{cases} \alpha_{\text{fast}}, & t \le T_{\text{warmup}}, \\ \dfrac{\alpha_{\text{fast}}}{1 + \frac{\alpha_{\text{fast}}/\alpha_{\text{slow}} - 1}{T_{\text{warmup}}}(t - T_{\text{warmup}})}, & T_{\text{warmup}} < t \le 2T_{\text{warmup}}, \\ \alpha_{\text{slow}}, & t > 2T_{\text{warmup}}, \end{cases}$$

where T_{warmup} is the warmup iterations for the learning rate.

Models. We utilize two popular classes of LLM architectures in the experiments:

- **Dense.** We adopt a dense decoder-only Transformer architecture that uses Rotary Positional Encoding (RoPE) (Su et al., 2024), SwiGLU feed-forward layers, RMSNorm, and a Pre-Norm design. Multi-head attention (MHA) is employed in self attention. Detailed configurations are provided in Table 1.
- **MoE.** The Mixture-of-Experts (MoE) architecture we use is built on a decoder-only Transformer with Multi-head Latent Attention (MLA) (DeepSeek-AI et al., 2024). Each MoE layer employs one shared expert together with partially activated experts per token. The first layer is configured as dense. We use an auxiliary-loss-free load balancing strategy. RMSNorm with a Pre-Norm design and RoPE (applied within the decoupled rotary components of MLA) are used throughout. Detailed configurations are provided in Table 2.

Table 1: Dense LLaMA model configurations. All models are trained with a default sequence length of 1024. The peak learning rate of MuonM is set to approximately 2/3 of that of MuonS without extra search.

Acronym	Size	d_{model}	d_{FFN}	n_{head}	n_{layer}	Batch size	lr_max		
							Muon	MuonS	MuonM
Dense (0.12B)	0.12B	768	2048	12	6	480	5e-3	1e-2	7e-3
Dense (0.25B)	0.25B	768	2048	16	24	512	3e-3	7e-3	4.5e-3
Dense (0.6B)	0.6B	1280	3412	20	24	512	1e-3	5e-3	3.5e-3
Dense (1.4B)	1.4B	2048	5461	32	24	512	7e-4	3e-3	2e-3

Learning Rate Schedulers. We evaluate two common learning rate schedules with $\min\{1000, T/50\}$ warm-up steps in all experiments (T is the total steps):

Table 2: MoE model configurations. All models are trained with a default sequence length of 4096 and batch size of 1024. The peak learning rate of MuonM is set to approximately 2/3 of that of MuonS without extra search.

Acronym	Size	Activated Size	d_{model}	n_{head}	n_{layer}	d_{FFN}			n_{expert}	MLA Dimensions					lr_max		
						Dense	Routed Expert	Shared Expert		r_q	r_{kv}	d_{qk}	d_{qk}^{rope}	d_v	Muon	MuonS	MuonM
MoE-0.6B	0.64B	0.13B	704	11	12	3584	672	672	4/32	704	240	128	64	128	7e-3	1e-2	7e-3
MoE-2.3B	2.27B	0.36B	1280	16	12	7168	704	1536	4/64	1536	512	128	64	128	3e-3	7e-3	4.5e-3
MoE-5.1B	5.06B	0.47B	1536	24	13	8192	1056	1152	4/80	512	256	128	64	128	1e-3	5e-3	3.5e-3

- `cos` (cosine decay): a linear warm-up to peak `lr_max`, followed by cosine decay to a terminal learning rate $\texttt{lr_min} = 0.05 \times \texttt{lr_max}$.
- `wsd` (warmup-stable-decay): a linear warm-up to `lr_max`, followed by a stable phase with constant learning rate of `lr_max` (up to 70% of total steps), and then a minus-sqrt $(1 - \sqrt{t})$ decay to $\texttt{lr_min} = 0.05 \times \texttt{lr_max}$ over the final 30% steps.

Table 3: Optimal peak learning rates for Muon and MuonS at various training steps on the 0.12B dense model. The peak learning rate of MuonM is set to approximately 2/3 of that of MuonS without extra search.

Iterations	20k	35k	50k	75k	100k	150k	200k
Muon	5e-3	3e-3	2e-3	2e-3	2e-3	2e-3	1e-3
MuonS	1e-2	1e-2	7e-3	7e-3	7e-3	7e-3	7e-3
MuonM	7e-3	7e-3	4.5e-3	4.5e-3	4.5e-3	4.5e-3	4.5e-3

D Additional Experimental Results

Block-wise RMS of Updates Figure 12 indicates that the RMS $\frac{1}{\sqrt{mn}}\|U_k\|_F$ for U_k in (4) remains stable for each transformer weight matrix. On the other hand, we use an initialization variance of 0.02^2, implying $R \approx 0.02$. Consequently, the operation $\mathsf{Norm}_R(U_k)$ is equivalent to multiplying by a factor nearly equal to 1. We therefore omit the re-normalization of U_k for brevity.

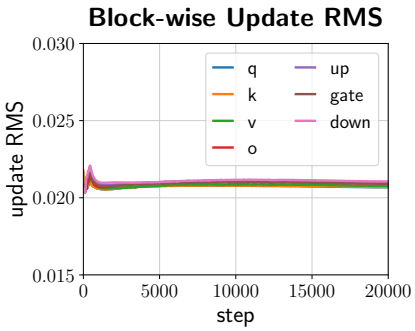

Figure 12: Block-wise RMS of the updates U_k in MuonM for the transformer weight matrices in the third layer of a 0.12B dense model.

Results with WSD Learning Rate Schedules Figure 13 shows the loss trajectories of Muon, MuonS, and MuonM under the WSD learning rate schedule. MuonM achieves the lowest terminal loss. The peak learning rates for Muon, MuonS, and MuonM are 0.005, 0.007 and 0.0045, respectively.

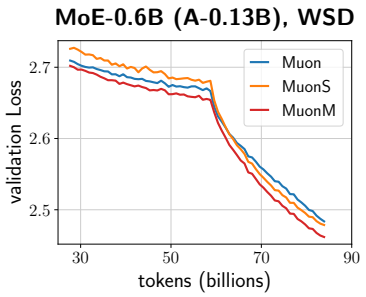

Figure 13: Validation losses of different optimizers on the 0.6B MoE model under the WSD schedule.

Hyper-parameters Sensitivity MuonM is relatively robust to the sharp-subspace dimension and to the choices of $\chi, \alpha_{\text{slow}}$ within a reasonable range, consistently achieving significant acceleration. On the Dense-0.12B model, varying the sharp-subspace dimension from 10% to 20% and 30% changes the loss only from 3.013 to 3.015 and 3.019, respectively, all well below MuonS (the un-accelerated baseline) at 3.033. Fixing the dimension at 10%, Figure 14 shows that all $\chi \in \{0.1, 0.2, 0.3\}$ and $\alpha_{\text{slow}} \in \{0.0005, 0.001, 0.002\}$ reduce the loss by more than 0.01 over MuonS, confirming the robustness.

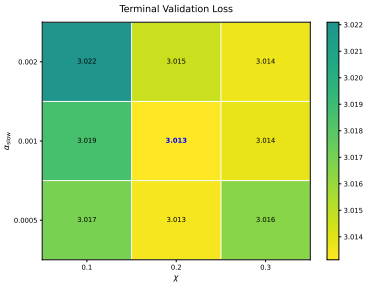

Figure 14: Effect of varying $(\chi, \alpha_{\text{slow}})$ on the terminal loss of the Dense-0.12B model.

Preconditioning for the Slow Momentum M^{slow} We do not design a specialized preconditioner for M^{slow}. Instead, we simply adopt row-wise normalization (RowScale in Algorithm 1). Directly applying the preconditioner of M^{fast} (i.e., $M^{\text{slow}}\left((\widetilde{M}^{\text{fast}})^\top \widetilde{M}^{\text{fast}}\right)^{-1/2}$) leads to numerical instability. On the other hand, applying an msign operation to M^{slow} similar to that for M^{fast} (i.e., replacing RowScale with msign in Algorithm 1) yields negligible loss improvement over simple row-wise normalization (Figure 15), while incurring additional computation and communication overhead in large-scale distributed training. We therefore adopt the simple row-wise normalization and leave further preconditioner design for M^{slow} as future work.

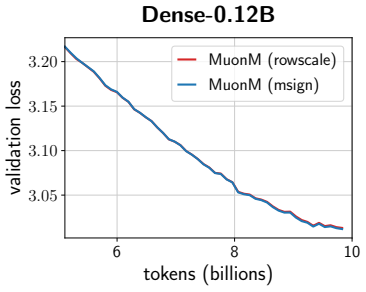

Figure 15: Comparison of using different preconditioning for M^{slow} on the Dense-0.12B model.

Comparison with Other Flat-Direction Enhancement Methods We compare MuonM with existing methods that also accelerate training dynamics along flat directions on the Dense-0.12B model. For Wang et al. (2025), we searched the learning rate ratios for the embedding , QK, VO, and FFN modules over $\{1, 2, 4\}$. The best configuration uses a ratio of $2 : 1 : 1 : 1$, yielding a terminal loss of 3.040, which is only 0.001 lower than Muon (compared with a reduction of 0.028 achieved by MuonM). For Zhou et al. (2025) and Wang et al. (2024), we were unable to run them with their default hyper-parameters due to out-of-memory errors caused by their optimizer states. Overall, MuonM achieves a larger loss reduction than these methods

E A Continuous-Time Perspective for Unifying Enhanced Momentum Methods

In this section, we establish the connections among AdEMAMix, GPA, EMA-Nesterov, SODA, and our method: all share the same continuous-time ODE form when preconditioning is ignored. For GPA and EMA-Nesterov, we assume the underlying base optimizer is Heavy Ball momentum. For brevity, our ODE derivation is conducted under gradient-noise-free conditions and with constant hyper-parameters.

Let $g_k = \nabla f(w_k)$ denote the gradient and P_k the preconditioner at the k-th iteration. Let $h > 0$ denote the discretization interval and $t_k = kh$. For the EMA coefficient ρ, we use the scaling $\rho = e^{-h/\tau} \approx 1 - \frac{h}{\tau}$ where $\tau > 0$ is the corresponding time constant. The algorithmic details and continuous limits are as follows.

Our Approach We illustrate our approach using MuonM as an example. Omitting the sphere constraint and Muon preconditioning yields the following discrete-time recursion (constants have been absorbed into step sizes for notational simplicity):

$$\begin{aligned} m_k^{\#} &= (1-\alpha_{\#})m_{k-1}^{\#} + \alpha_{\#} g_k, \quad \# \in \{\text{fast}, \text{slow}\}, \\ w_{k+1} &= w_k - \eta_k\big(\chi m_k^{\text{slow}} + \gamma g_k + m_k^{\text{fast}}\big). \end{aligned} \tag{14}$$

Under a minor change of notation, the continuous-time counterpart takes the form

$$\begin{aligned} \dot{m}_{\text{f}} &= -\tau_{\text{f}}^{-1} m_{\text{f}} + \nabla f(w), \\ \dot{m}_{\text{s}} &= -\tau_{\text{s}}^{-1} m_{\text{s}} + \nabla f(w), \\ \dot{w} &= -a_{\text{f}} m_{\text{f}} - a_{\text{s}} m_{\text{s}} - a_{\text{g}} \nabla f(w). \end{aligned} \tag{15}$$

Let $g_t = \nabla f(w_t)$. The momentum equations can be written as $\big(1 + \tau_{\text{f}} \frac{d}{dt}\big) m_{\text{f}} = \tau_{\text{f}} g_t$, $\big(1 + \tau_{\text{s}} \frac{d}{dt}\big) m_{\text{s}} = \tau_{\text{s}} g_t$.

Applying the operator $\big(1 + \tau_{\text{f}} \frac{d}{dt}\big)\big(1 + \tau_{\text{s}} \frac{d}{dt}\big)$ to $\frac{d}{dt} w_t = -a_{\text{f}} m_{\text{f}} - a_{\text{s}} m_{\text{s}} - a_{\text{g}} g_t$, we obtain

$$\begin{aligned} 0 = &\left(1 + \tau_{\text{f}} \frac{d}{dt}\right)\left(1 + \tau_{\text{s}} \frac{d}{dt}\right) \frac{d}{dt} w_t + a_{\text{f}} \tau_{\text{f}} \left(1 + \tau_{\text{s}} \frac{d}{dt}\right) g_t \\ &+ a_{\text{s}} \tau_{\text{s}} \left(1 + \tau_{\text{f}} \frac{d}{dt}\right) g_t + a_{\text{g}} \left(1 + \tau_{\text{f}} \frac{d}{dt}\right)\left(1 + \tau_{\text{s}} \frac{d}{dt}\right) g_t. \end{aligned}$$

Expanding the operator products gives the third-order ODE

$$\begin{aligned} 0 = &\tau_{\text{f}} \tau_{\text{s}} \frac{d^3}{dt^3} w_t + (\tau_{\text{f}} + \tau_{\text{s}}) \frac{d^2}{dt^2} w_t + \frac{d}{dt} w_t + (a_{\text{f}} \tau_{\text{f}} + a_{\text{s}} \tau_{\text{s}} + a_{\text{g}}) \nabla f(w_t) \\ &+ \left[\tau_{\text{f}} \tau_{\text{s}} (a_{\text{f}} + a_{\text{s}}) + a_{\text{g}} (\tau_{\text{f}} + \tau_{\text{s}})\right] \nabla^2 f(w_t) \frac{d}{dt} w_t \\ &+ a_{\text{g}} \tau_{\text{f}} \tau_{\text{s}} \left(\nabla^3 f(w_t) \left[\frac{d}{dt} w_t, \frac{d}{dt} w_t\right] + \nabla^2 f(w_t) \frac{d^2}{dt^2} w_t\right), \end{aligned}$$

where we also use

$$\frac{d^2}{dt^2} \nabla f(w_t) = \nabla^3 f(w_t) \left[\frac{d}{dt} w_t, \frac{d}{dt} w_t\right] + \nabla^2 f(w_t) \frac{d^2}{dt^2} w_t.$$

GPA GPA maintains a base sequence z_k, an exponentially averaged sequence x_k, and an interpolation point y_k at which the stochastic gradient is evaluated (Defazio et al., 2026). The generic GPA framework permits an arbitrary base optimizer. We instantiate it with preconditioned Heavy Ball:

$$\begin{aligned} y_k &= \mu_y x_k + (1-\mu_y) z_k, \\ b_k &= \rho_{\mathrm{H}} b_{k-1} + (1-\rho_{\mathrm{H}}) g_k, \\ d_k &= P_k b_k, \\ z_{k+1} &= (1-\eta_k \lambda) z_k - \eta_k d_k, \\ x_{k+1} &= \mu_x x_k + (1-\mu_x) z_{k+1}. \end{aligned} \tag{16}$$

Here b_k is the Heavy-Ball state, μ_x controls primal averaging, and μ_y controls the interpolation between the averaged and base iterates.

The continuous time limit of (16) without preconditioner and weight decay can be written as

$$\begin{aligned} \dot{b} &= -\tau_{\mathrm{f}}^{-1} b + \tau_{\mathrm{f}}^{-1} \nabla f(y), \\ \dot{z} &= -b, \\ \dot{x} &= -\tau_{\mathrm{s}}^{-1} x + \tau_{\mathrm{s}}^{-1} z, \\ y &= \mu_y x + (1-\mu_y) z. \end{aligned} \tag{17}$$

The last two equations imply

$$(1 + \tau_{\mathrm{s}} \frac{d}{dt}) y = \big(1 + (1-\mu_y)\tau_{\mathrm{s}} \frac{d}{dt}\big) z. \tag{18}$$

The first two equations give

$$(1 + \tau_{\mathrm{f}} \frac{d}{dt}) \frac{d}{dt} z + \nabla f(y) = 0. \tag{19}$$

Eliminating z yields the third-order GPA (Heavy-Ball) ODE

$$(1 + \tau_{\mathrm{f}} \frac{d}{dt})(1 + \tau_{\mathrm{s}} \frac{d}{dt}) \frac{d}{dt} y + \big(1 + (1-\mu_y)\tau_{\mathrm{s}} \frac{d}{dt}\big) \nabla f(y) = 0. \tag{20}$$

Namely

$$\tau_{\mathrm{f}} \tau_{\mathrm{s}} \frac{d^3}{dt^3} y_t + \big(\tau_{\mathrm{f}} + \tau_{\mathrm{s}}\big) \frac{d^2}{dt^2} y_t + \frac{d}{dt} y_t + \nabla f(y_t) + (1-\mu_y)\tau_{\mathrm{s}} \nabla^2 f(y_t) \frac{d}{dt} y_t = 0. \tag{21}$$

AdEMAMix AdEMAMix augments AdamW with a second, substantially slower first-moment EMA (Pagliardini et al., 2025). Let m_k^{f} and m_k^{s} denote the fast and slow first moments, and let v_k denote the second moment. Its complete update is

$$\begin{aligned} m_k^{\mathrm{f}} &= \beta_1 m_{k-1}^{\mathrm{f}} + (1-\beta_1) g_k, & \widehat{m}_k^{\mathrm{f}} &= \frac{m_k^{\mathrm{f}}}{1-\beta_1^k}, \\ m_k^{\mathrm{s}} &= \beta_{3,k} m_{k-1}^{\mathrm{s}} + (1-\beta_{3,k}) g_k, & & \\ v_k &= \beta_2 v_{k-1} + (1-\beta_2) g_k^{\odot 2}, & \widehat{v}_k &= \frac{v_k}{1-\beta_2^k}, \\ w_{k+1} &= (1-\eta_k\lambda) w_k - \eta_k \frac{\widehat{m}_k^{\mathrm{f}} + a_k m_k^{\mathrm{s}}}{\sqrt{\widehat{v}_k} + \epsilon}. & & \end{aligned} \tag{22}$$

After removing the second-moment preconditioner, bias-correction and weight decay, the continuous state equations are

$$\begin{aligned} \dot{m}_{\mathrm{f}} &= -\tau_{\mathrm{f}}^{-1} m_{\mathrm{f}} + \tau_{\mathrm{f}}^{-1} \nabla f(w), \\ \dot{m}_{\mathrm{s}} &= -\tau_{\mathrm{s}}^{-1} m_{\mathrm{s}} + \tau_{\mathrm{s}}^{-1} \nabla f(w), \\ \dot{w} &= -a_{\mathrm{f}} m_{\mathrm{f}} - a_{\mathrm{s}} m_{\mathrm{s}}. \end{aligned} \tag{23}$$

Applying $(1 + \tau_{\mathrm{f}} \frac{d}{dt})(1 + \tau_{\mathrm{s}} \frac{d}{dt})$ to the parameter equation and eliminating the two momentum states gives

$$(1 + \tau_{\mathrm{f}} \frac{d}{dt})(1 + \tau_{\mathrm{s}} \frac{d}{dt}) \frac{d}{dt} w + a_{\mathrm{f}} (1 + \tau_{\mathrm{s}} \frac{d}{dt}) \nabla f(w) + a_{\mathrm{s}} (1 + \tau_{\mathrm{f}} \frac{d}{dt}) \nabla f(w) = 0. \tag{24}$$

Equivalently,

$$\tau_{\mathrm{f}} \tau_{\mathrm{s}} \frac{d^3}{dt^3} w_t + \big(\tau_{\mathrm{f}} + \tau_{\mathrm{s}}\big) \frac{d^2}{dt^2} w_t + \frac{d}{dt} w_t + (a_{\mathrm{f}} + a_{\mathrm{s}}) \nabla f(w_t) + (a_{\mathrm{f}}\tau_{\mathrm{s}} + a_{\mathrm{s}}\tau_{\mathrm{f}}) \nabla^2 f(w_t) \frac{d}{dt}(w_t) = 0. \tag{25}$$

EMA-Nesterov EMA-Nesterov is a wrapper that replaces the one-step Nesterov look-ahead direction by an EMA of parameter increments (Yau et al., 2026). Let z_k denote the EMA look-ahead state. When the wrapped base optimizer is preconditioned Heavy Ball momentum, the update rule is

$$\begin{aligned} y_k &= x_k + \beta_k z_k, \\ b_k &= \rho_{\mathrm{H}} b_{k-1} + (1-\rho_{\mathrm{H}}) g_k, \\ x_{k+1} &= (1-\eta_k\lambda) y_k - \eta_k P_k b_k, \\ z_{k+1} &= \rho_z z_k + (1-\rho_z)(x_{k+1} - x_k). \end{aligned} \tag{26}$$

If $\mathcal{A}_k$ denotes the base-optimizer map, the original wrapper is $x_{k+1} = \mathcal{A}_k(x_k + \beta_k z_k)$ followed by the final line of (26).

The continuous time limit of (26) without preconditioning and weight decay is

$$\begin{aligned} \dot{b} &= -\tau_{\mathrm{f}}^{-1} b + \tau_{\mathrm{f}}^{-1} \nabla f(x), \\ \dot{z} &= -\tau_{\mathrm{s}}^{-1} z + \tau_{\mathrm{s}}^{-1} \dot{x}, \\ \dot{x} &= \beta z - b. \end{aligned} \tag{27}$$

Eliminating b and z gives

$$(1 + \tau_{\mathrm{f}} \frac{d}{dt}) \left[(1 + \tau_{\mathrm{s}} \frac{d}{dt}) \frac{d}{dt} - \beta \frac{d}{dt} \right] x + (1 + \tau_{\mathrm{s}} \frac{d}{dt}) \nabla f(x) = 0. \tag{28}$$

Equivalently,

$$\tau_{\mathrm{f}} \tau_{\mathrm{s}} \frac{d^3}{dt^3} x_t + \big(\tau_{\mathrm{f}} + \tau_{\mathrm{s}} - \beta\tau_{\mathrm{f}}\big) \frac{d^2}{dt^2} x_t + \big(1 - \beta\big) \frac{d}{dt} x_t + \nabla f(x_t) + \tau_{\mathrm{s}} \nabla^2 f(x_t) \frac{d}{dt} x_t = 0. \tag{29}$$

SODA SODA combines gradient averaging, dual optimism, primal averaging, and primal extrapolation (Pethick et al., 2026). Its full update is

$$\begin{aligned} m_{k+1} &= (1-\alpha_k) m_k + \alpha_k \nabla f(y_k), \\ \bar{m}_{k+1} &= (1-\bar{\alpha}_k) m_{k+1} + \bar{\alpha}_k \nabla f(y_k), \\ z_{k+1} &\in \partial h_k^*(-\gamma_k \bar{m}_{k+1}) = \arg\min_z \left\{ \gamma_k \langle \bar{m}_{k+1}, z \rangle + h_k(z) \right\}, \\ x_{k+1} &= (1-\lambda_k) x_k + \lambda_k z_{k+1}, \\ y_{k+1} &= (1-\bar{\lambda}_k) x_{k+1} + \bar{\lambda}_k z_{k+1}. \end{aligned} \tag{30}$$

We consider the Euclidean mirror map $h(z) = \frac{1}{2}\|z\|^2$, and freeze these coefficients with new notations. The continuous limit is

$$\begin{aligned} \dot{m} &= -\tau^{-1} m + \tau^{-1} \nabla f(y), \\ \dot{x} &= -\lambda_1 x - \eta(m + \omega \nabla f(y)), \\ y &= (1-\lambda_2) x - \nu(m + \omega \nabla f(y)). \end{aligned} \tag{31}$$

Define the quantities $s = m + \omega g(y), \delta = \eta a + \nu\lambda_1$. The state equations then become

$$\dot{x} = -\lambda_1 x - \eta s, \qquad y = (1-\lambda_2) x - \nu s. \tag{32}$$

Moreover,

$$\dot{s} = \dot{m} + \omega \nabla^2 f(y) \dot{y} = -\tau^{-1} s + \frac{1+\omega}{\tau} \nabla f(y) + \omega \nabla^2 f(y) \dot{y}. \tag{33}$$

Differentiating $y = ax - \nu s$ and using (32)–(33) yield

$$\big(I + \nu\omega \nabla^2 f(y)\big) \dot{y} = -\lambda_1 y + q s - \frac{\nu(1+\omega)}{\tau} g(y), \tag{34}$$

where $q = \nu\big(\tau^{-1} - \lambda_1\big) - \eta(1-\lambda_2)$.

Differentiating (34) and eliminating s and $\dot{s}$ give a closed equation for y:

$$\big(I + \nu\omega\nabla^2 f(y)\big)\ddot{y} + \nu\omega\nabla^3 f(y)[\dot{y},\dot{y}] + \left[\big(\lambda_1 + \frac{1}{\tau}\big)I + \ell\nabla^2 f(y)\right]\dot{y} + \frac{\lambda_1}{\tau}y + \frac{(1+\omega)\delta}{\tau}\nabla f(y) = 0, \tag{35}$$

where $\ell = \frac{\nu(1+\omega)}{\tau} + \omega\delta$, and the $\frac{\lambda_1}{\tau}y$ term denotes the weight decay. Optimistic momentum and primal extrapolation together yield an implicit preconditioner $\big(I + \nu\omega\nabla^2 f(y)\big)^{-1}$.

E.1 Unified ODE Representation

The preceding derivation indicates that these continuous-time limits can be cast in a unified form

$$\mathcal{A}\left(\frac{d}{dt}\right) w_t + \mathcal{B}\left(\frac{d}{dt}\right)\nabla f(w_t) = 0, \tag{36}$$

where $\mathcal{A}, \mathcal{B}$ are matrix-valued polynomials in the differential operator $\frac{d}{dt}$:

$$\mathcal{A}\left(\frac{d}{dt}\right) = \sum_{i=0}^{n} A_i \left(\frac{d}{dt}\right)^i, \qquad \mathcal{B}\left(\frac{d}{dt}\right) = \sum_{i=0}^{m} B_i \left(\frac{d}{dt}\right)^i. \tag{37}$$

For two-timescale momentum, $\mathcal{A}$ is of degree three. This polynomial framework naturally generalizes to the n-timescale case, where the degree of $\mathcal{A}$ becomes $n+1$.

F Illustrative Theoretical Analysis

F.1 Variance Reduction of Multiscale Momentum

In this section, we analyze the variance of multiscale momentum through an illustrative Gaussian example, demonstrating the variance-reduction effect of introducing a slow momentum component. Consider

$$\begin{aligned} m_k^{\text{slow}} &= (1-\alpha_{\text{slow}})\, m_{k-1}^{\text{slow}} + \alpha_{\text{slow}}\, g_k, \\ m_k^{\text{fast}} &= (1-\alpha_{\text{fast}})\, m_{k-1}^{\text{fast}} + \alpha_{\text{fast}}\, g_k, \\ m_k &= c\, m_k^{\text{slow}} + (1-c)\, m_k^{\text{fast}}, \end{aligned} \tag{38}$$

where $0 < \alpha_{\text{slow}} < \alpha_{\text{fast}} \le 1$ and $g_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu, \Sigma^2)$. We are interested in the stationary covariance of the update m_k which preserves the stationary mean, i.e., $\mathbb{E}[m_\infty] = \mu$.

Let $\epsilon_k = g_k - \mu$ denote the zero-mean noise. Unrolling the recursions, the noise components of the two momenta are

$$m_k^{\text{slow}} - \mu = \sum_{j=0}^{k} (1-\alpha_{\text{slow}})^j\, \alpha_{\text{slow}}\, \epsilon_{k-j}, \qquad m_k^{\text{fast}} - \mu = \sum_{j=0}^{k} (1-\alpha_{\text{fast}})^j\, \alpha_{\text{fast}}\, \epsilon_{k-j}. \tag{39}$$

Therefore, in stationarity,

$$\begin{aligned} &\mathrm{Cov}(m_\infty^{\text{slow}}) = \sum_{j=0}^{\infty} (1-\alpha_{\text{slow}})^{2j}\, \alpha_{\text{slow}}^2\, \Sigma^2, \qquad \mathrm{Cov}(m_\infty^{\text{fast}}) = \sum_{j=0}^{\infty} (1-\alpha_{\text{fast}})^{2j}\, \alpha_{\text{fast}}^2\, \Sigma^2, \\ &\mathrm{Cov}(m_\infty^{\text{slow}}, m_\infty^{\text{fast}}) = \sum_{j=0}^{\infty} (1-\alpha_{\text{slow}})^{j} (1-\alpha_{\text{fast}})^{j}\, \alpha_{\text{slow}}\alpha_{\text{fast}}\, \Sigma^2. \end{aligned} \tag{40}$$

Summing the geometric series, it follows that

$$\mathrm{Cov}(m_\infty) = \left(\frac{c^2\, \alpha_{\text{slow}}^2}{1-(1-\alpha_{\text{slow}})^2} + \frac{(1-c)^2\, \alpha_{\text{fast}}^2}{1-(1-\alpha_{\text{fast}})^2} + \frac{2c(1-c)\, \alpha_{\text{slow}}\alpha_{\text{fast}}}{1-(1-\alpha_{\text{slow}})(1-\alpha_{\text{fast}})}\right)\Sigma^2. \tag{41}$$

Simplifying yields

$$\mathrm{Cov}(m_\infty) = \underbrace{\left(\frac{c^2\,\alpha_{\text{slow}}}{2-\alpha_{\text{slow}}} + \frac{(1-c)^2\,\alpha_{\text{fast}}}{2-\alpha_{\text{fast}}} + \frac{2c(1-c)\,\alpha_{\text{slow}}\alpha_{\text{fast}}}{\alpha_{\text{slow}}+\alpha_{\text{fast}}-\alpha_{\text{slow}}\alpha_{\text{fast}}} \right)}_{:=\,f(c)} \Sigma^2. \tag{42}$$

Differentiating with respect to c gives

$$f'(c) = -\frac{2(\alpha_{\text{fast}}-\alpha_{\text{slow}})\Big(\alpha_{\text{fast}}(2-\alpha_{\text{slow}}) - 2(\alpha_{\text{fast}}-\alpha_{\text{slow}})\,c\Big)}{(\alpha_{\text{slow}}+\alpha_{\text{fast}}-\alpha_{\text{slow}}\alpha_{\text{fast}})(2-\alpha_{\text{slow}})(2-\alpha_{\text{fast}})} < 0, \tag{43}$$

since every factor is positive for $0 < \alpha_{\text{slow}} < \alpha_{\text{fast}} \le 1$ and $c \in [0,1]$. Hence f is strictly decreasing on $[0,1]$. **Increasing the weight c of the slow momentum monotonically reduces the stationary variance.** In particular, the two limiting regimes are

$$f(0) = \frac{\alpha_{\text{fast}}}{2-\alpha_{\text{fast}}} \quad \text{(fast momentum only)}, \qquad f(1) = \frac{\alpha_{\text{slow}}}{2-\alpha_{\text{slow}}} \quad \text{(slow momentum only)}. \tag{44}$$

Thus incorporating slow momentum reduces the stationary variance by a factor of up to $\dfrac{\alpha_{\text{slow}}(2-\alpha_{\text{fast}})}{\alpha_{\text{fast}}(2-\alpha_{\text{slow}})} \approx \dfrac{\alpha_{\text{slow}}}{\alpha_{\text{fast}}}$ relative to fast momentum alone.

F.2 Weight Norm Inflation by Slow Momentum

In this section, we provide a theoretical understanding of the role of slow momentum in norm inflation. Motivated by the fact that noise dominates in LLM pretraining, the intuition behind norm inflation for a scale-invariant loss L (i.e., $L(cW) = L(W)$ for any $c > 0$) is that a Brownian-motion-like movement drives the norm to blow up, while weight decay can suppress it. We therefore approximately assume that the stochastic gradients $g_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2 I_d)$. For brevity, we assume that the projection onto the flat directions $\mathcal{Q}$ is constant across iterations and has rank $q \approx d$. We then consider the update with weight decay coefficient$\lambda > 0$

$$\begin{aligned} m_k^{\#} &= (1-\alpha_{\#})m_{k-1}^{\#} + g_k, \quad \# \in \{\text{fast}, \text{slow}\}, \\ w_{k+1}^{\chi} &= (1-\eta\lambda)w_k^{\chi} - \eta(\chi\mathcal{Q}m_k^{\text{slow}} + m_k^{\text{fast}}), \end{aligned} \tag{45}$$

where $m_{-1}^{\text{fast}} = m_{-1}^{\text{slow}} = 0$ and the superscript χ denotes the slow momentum coefficient. We have the following conclusion.

Proposition F.1 (Slow-momentum-induced norm inflation)**.** *Suppose $\rho := 1-\eta\lambda > 0$ and both processes start from the same deterministic w_0. Then for every $k \ge 0$, the iterates in* (45) *satisfy*

$$\mathbb{E}\|w_{k+1}^{\chi}\|^2 - \mathbb{E}\|w_{k+1}^{0}\|^2 = q\eta^2\sigma^2 \sum_{n=0}^{k} \left(2\chi A_{\text{fast},n}A_{\text{slow},n} + \chi^2 A_{\text{slow},n}^2\right), \tag{46}$$

where $\beta_{\text{fast}} = 1-\alpha_{\text{fast}}, \beta_{\text{slow}} = 1-\alpha_{\text{slow}}$ and

$$A_{\#,n} := \sum_{\ell=0}^{n} \rho^{\,n-\ell}\beta_{\#}^{\ell}, \qquad \# \in \{\text{fast}, \text{slow}\}.$$

Moreover, the weight norm converges to

$$\lim_{k\to\infty} \mathbb{E}\|w_k^{\chi}\|^2 = V_\chi := d\eta^2\sigma^2 \sum_{n=0}^{\infty} A_{\text{fast},n}^2 + \underbrace{q\eta^2\sigma^2 \sum_{n=0}^{\infty} \left(2\chi A_{\text{fast},n}A_{\text{slow},n} + \chi^2 A_{\text{slow},n}^2\right)}_{\textit{inflated norm by slow momentum}}, \tag{47}$$

where $V_\chi < \infty$ if and only if $\rho < 1$ (i.e., $\lambda > 0$).

Remark F.2. The norm inflation due to slow momentum has two contributing factors. The first is the explicit positive coefficient $\chi > 0$, and the second is the asymptotic dominance $A_{\text{slow},n} \gg A_{\text{fast},n}$ for large n.

Proof. Unrolling the momentum recursions gives

$$m_k^{\#} = \sum_{j=0}^{k} \beta_{\#}^{k-j} g_j, \qquad \# \in \{\text{fast}, \text{slow}\}.$$

Substituting it into the update rule yields

$$w_{k+1}^{\chi} = \rho^{k+1} w_0 - \eta \sum_{j=0}^{k} \left(A_{\text{fast},k-j} I_d + \chi A_{\text{slow},k-j} \mathcal{Q} \right) g_j.$$

Since $g_k \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2 I_d)$,

$$\mathbb{E}\|w_{k+1}^{\chi}\|^2 = \rho^{2(k+1)} \|w_0\|^2 + \eta^2 \sigma^2 \sum_{j=0}^{k} \|A_{\text{fast},k-j} I_d + \chi A_{\text{slow},k-j} \mathcal{Q}\|_F^2 .$$

We obtain (46) by

$$\|A_{\text{fast},n} I_d + \chi A_{\text{slow},n} \mathcal{Q}\|_F^2 - \|A_{\text{fast},n} I_d\|_F^2 = q \left(2\chi A_{\text{fast},n} A_{\text{slow},n} + \chi^2 A_{s,n}^2 \right).$$

Moreover,

$$A_{\#,n} = \begin{cases} \dfrac{\rho^{n+1} - \beta_{\#}^{n+1}}{\rho - \beta_{\#}}, & \rho \neq \beta_{\#}, \\ (n+1)\rho^n, & \rho = \beta_{\#}, \end{cases}$$

gives $\sum_{n=0}^{\infty} A_{\#,n}^2 < \infty$ iff $\lambda > 0$. Therefore, the stationary mean-square parameter norms are finite iff $\lambda > 0$, and taking the trace of the stationary covariance matrix gives V_{χ} in (47). □

G Discussions

G.1 On the Complementary Design of Preconditioning and Momentum for Enhancing Training Dynamics along Flat Directions

Momentum and preconditioning play distinct yet synergistic roles in accelerating training dynamics. Preconditioning modulates the learning rate along different eigen-directions. A preconditioner that more accurately approximates the Hessian can more precisely tune the learning rates along directions of different curvature, and in particular, amplify the effective learning rate along flat directions. However, adjusting the learning rate alone proportionally amplifies both the true signal and the noise in the momentum. This study therefore focuses on momentum design to achieve variance-reduced acceleration of flat-direction training dynamics.

Since pretraining is noise-dominated and exhibits noise–Hessian alignment, we consider an idealized adaptive update as an illustrative example:

$$u = \mathbb{E}[gg^\top]^{-\alpha} g,$$

where $g \in \mathbb{R}^d$ is the stochastic gradient, α is the power and $\mathbb{E}[gg^\top]$ serves as a proxy for the dominant Hessian component under noise-Hessian alignment. The covariance of u becomes approximately $\text{Cov}(u) \approx \mathbb{E}[gg^\top]^{1-2\alpha}$. Modern adaptive optimizers approximate this with $\alpha = 1/2$, which achieves a favorable preconditioning–stability trade-off: the covariance of u is nearly identity, while the ill-conditioned curvature is alleviated to some extent. A larger α allows faster progress along flat directions but also amplifies the tail noise. The training stability becomes difficult to control. Therefore, improving the preconditioner alone cannot achieve variance-reduced flat-direction acceleration. This motivates our focus on the momentum axis, which directly reduces variance along flat directions and is complementary to preconditioner design.

G.2 Understanding Robust Variance Reduction of Flat-Direction Multiscale Momentum: A Bias-Variance Trade-off View

Recall the flat-direction multiscale momentum method in (2):

$$\begin{aligned} m_k^{\#} &= (1-\alpha_{\#})m_{k-1}^{\#} + \alpha_{\#} g_k, \quad \# \in \{\text{fast}, \text{slow}\}, \\ w_{k+1} &= w_k - \eta_k\big(c\mathcal{Q}_k m_k^{\text{slow}} + (1-c)m_k^{\text{fast}}\big) \\ &= w_k - \eta_k\big(\mathcal{Q}_k(cm_k^{\text{slow}} + (1-c)m_k^{\text{fast}}) + (1-c)\mathcal{P}_k m_k^{\text{fast}}\big), \end{aligned} \tag{48}$$

where $\alpha_{\text{fast}} \gg \alpha_{\text{slow}} > 0$ denote the momentum decay rates, $0 < c < 1$ is the interpolation coefficient, and $\mathcal{P}_k$, $\mathcal{Q}_k$ are the projections onto sharp and flat subspaces.

Assume $m_k^{\text{fast}} = \nabla f(w_k) + b_k^{\text{fast}} + \xi_k^{\text{fast}}$, $m_k^{\text{slow}} = \nabla f(w_k) + b_k^{\text{slow}} + \xi_k^{\text{slow}}$ with bias $\|b_k^{\text{slow}}\| \gg \|b_k^{\text{fast}}\|$ (due to slower forgetting over longer gradient histories), zero mean noise $\|\xi_k^{\text{slow}}\| \ll \|\xi_k^{\text{fast}}\|$ and noise dominating the fast momentum such that $\|\xi_k^{\text{fast}}\| \gg \|b_k^{\text{fast}}\| + \|\nabla f(w_k)\|$. Then, for an appropriate choice of c, the expected loss decrease conditioned on w_k resulting from the $\mathcal{Q}_k$-subspace update can be approximated via a second-order Taylor expansion:

$$\begin{aligned} &\mathbb{E}_k[f(w_{k+1}) - f(w_k)]|_{\mathcal{Q}_k} \\ \approx& -\eta\mathbb{E}_k\left\langle \mathcal{Q}_k\nabla f(w_k), c\mathcal{Q}_k m_k^{\text{slow}} + (1-c)\mathcal{Q}_k m_k^{\text{fast}}\right\rangle \\ &+ \frac{\eta^2}{2}\mathbb{E}_k(cm_k^{\text{slow}} + (1-c)m_k^{\text{fast}})^\top \mathcal{Q}_k\nabla^2 f(w_k)\mathcal{Q}_k(cm_k^{\text{slow}} + (1-c)m_k^{\text{fast}}) \\ \approx& \underbrace{-\eta\|\mathcal{Q}_k\nabla f(w_k)\|^2}_{\text{gradient descent}} + \underbrace{c\eta\left\langle \mathcal{Q}_k\nabla f(w_k), -\mathcal{Q}_k b_k^{\text{slow}}\right\rangle}_{\text{bias injection}} + \underbrace{\frac{(1-c)^2\eta^2}{2}\mathbb{E}_k(\xi_k^{\text{fast}})^\top \mathcal{Q}_k\nabla^2 f(w_k)\mathcal{Q}_k\xi_k^{\text{fast}}}_{\text{noise variance}}. \end{aligned}$$

The noise-variance term is typically positive, since noise tends to accumulate along directions with larger Hessian eigenvalues (Wu et al., 2022; Song et al., 2025; Wang et al., 2026b). Assuming the bias $\mathcal{Q}_k b_k^{\text{slow}}$ in $\mathcal{Q}_k$ subspace is moderate relative to $\nabla f(w_k)$, we can choose suitable $c < 1$ so that the bias injection is negligible compared to the gradient-descent term, while the noise variance term is simultaneously suppressed, yielding an improved local loss reduction. For the $\mathcal{P}_k$ subspace, we note that $\mathbb{E}_k[f(w_{k+1}) - f(w_k)]|_{\mathcal{P}_k} \approx 0$ as training typically lies in the Edge of Stability regime (Cohen et al., 2021; Andreyev & Beneventano, 2025; Andreyev et al., 2026). On the other hand, take $c > 0$ also reduces the actual learning rate in the $\mathcal{P}_k$ subspace, which helps decrease oscillations. We exclude slow momentum from $\mathcal{P}_k$, as slow-momentum bias in sharp directions can be much larger than the gradient signal, which would force c to be vanishingly small and thereby restrict the acceleration in flat directions. Consequently, the overall effect is a net reduction in the local loss.

H Empirical Fitting of LLM Pre-training Loss Trajectory

In this section, we show that the theoretical asymptotic estimate of the excess risk in (8),

$$\mathbb{E}\mathcal{E}(w_k) \asymp \underbrace{\mathsf{S}(T_k)}_{\text{signal learning}} + \underbrace{\sigma^2\sum_{i=0}^{k-1}\eta_i^2\mathsf{K}(T_k - T_i)}_{\text{noise accumulation}}, \tag{49}$$

can closely approximate the actual loss trajectories of LLM pretraining.

Given a learning rate schedule $\{\eta_k\}$, we model the expected loss based on (8) as

$$L_k = L_\infty + c_{\text{signal}}\, S(T_k) + c_{\text{noise}}\sum_{i=k_0}^{k}\eta_i^2 K(T_k - T_i), \tag{50}$$

where k_0 is the fitting origin (the first update retained for fitting), L_∞ is a nonnegative loss floor, and S and K are the signal-learning and noise-accumulation kernels, respectively (which are defined below). The noise intensity factor σ^2 is absorbed into c_{noise}.

Table 4: The fitted results for MuonM.

Model	r	RMSE
Dense-0.25B	3	4.28×10^{-3}
Dense-0.6B	5	5.12×10^{-3}
Dense-1.4B	9	3.94×10^{-3}
MoE-0.6B (A-0.13B)	3	3.25×10^{-3}
MoE-2.3B (A-0.36B)	3	3.83×10^{-3}
MoE-5.1B (A-0.47B)	7	2.06×10^{-3}

Signal and forgetting kernels Let

$$T_k = T_0 + c_{\mathrm{lr}} \sum_{q=k_0}^{k} \eta_q,$$

where $T_0 > 0$ is a learnable time offset and $c_{\mathrm{lr}} > 0$ rescales the learning rate accumulation. For an integer $r \geq 0$ that determines the sharp subspace dimension, we set

$$S(t) = c_s \sum_{j=1}^{r} j^{-\nu}\, j^{-2-2(s-1)\nu}\, \exp\!\big(-2a_{\mathrm{top}} j^{-\nu} t\big) + \Big(C_{r,s}\,(r+1)^{\nu} + a_{\mathrm{full}}\, t\Big)^{-s}, \tag{51}$$

and

$$K(t) = c_k \sum_{j=1}^{r} a_{\mathrm{top}}^2\, j^{-2\nu}\, \exp\!\big(-2a_{\mathrm{top}} j^{-\nu} t\big) + a_{\mathrm{full}}^2 \Big(C_{r,k}\,(r+1)^{\nu} + a_{\mathrm{full}}\, t\Big)^{-\left(2-\nu^{-1}\right)}. \tag{52}$$

Here we treat r as learnable, and note that its value cannot directly match the actual dimension of the sharp subspace in real experiments. We enforce the ordering constraint $a_{\mathrm{full}} \geq a_{\mathrm{top}} > 0$ via the reparameterization $a_{\mathrm{full}} = a_{\mathrm{top}} + \mathrm{softplus}(\Delta_a)$.

Fitting objective We use logged validation-loss points after the fitting origin. Let $\mathcal{D}_{\mathrm{fit}}$ denote these points and $\widehat{L}_k$ denote the logged validation loss at iteration k. We fit the curve by minimizing the scaled log-Huber objective

$$\mathcal{J}(\theta) = \frac{1}{|\mathcal{D}_{\mathrm{fit}}|} \sum_{k \in \mathcal{D}_{\mathrm{fit}}} \rho_\delta\big(\log \widehat{L}_k - \log L_k\big), \tag{53}$$

with

$$\rho_\delta(u) = \begin{cases} \frac{1}{2}(u/\delta)^2, & |u| \leq \delta, \\ |u|/\delta - \frac{1}{2}, & |u| > \delta, \end{cases} \tag{54}$$

and default threshold $\delta = 10^{-3}$. Positive parameters are mapped through softplus (and L_∞ through a sigmoid upper-bounded by the observed minimum loss), so that unconstrained optimization automatically respects positivity.

Fitting hyper-parameters The integer cutoff r is treated as a discrete outer-loop parameter. For each candidate $r \in \{0, 1, 3, 5, 7, 9, 20, 50\}$, we optimize the continuous parameters by Adam (learning rate $= 0.035$) followed by BFGS (learning rate $= 0.1$ with Wolfe line search). After continuous optimization, we select the r with the best fitting loss. Fitting begins at 10% of total progress and continues through 100%.

Fitted results The fitting results are shown in Table 4 and Figures 16, 9, demonstrating that the fitted curve closely follows the true loss trajectory.

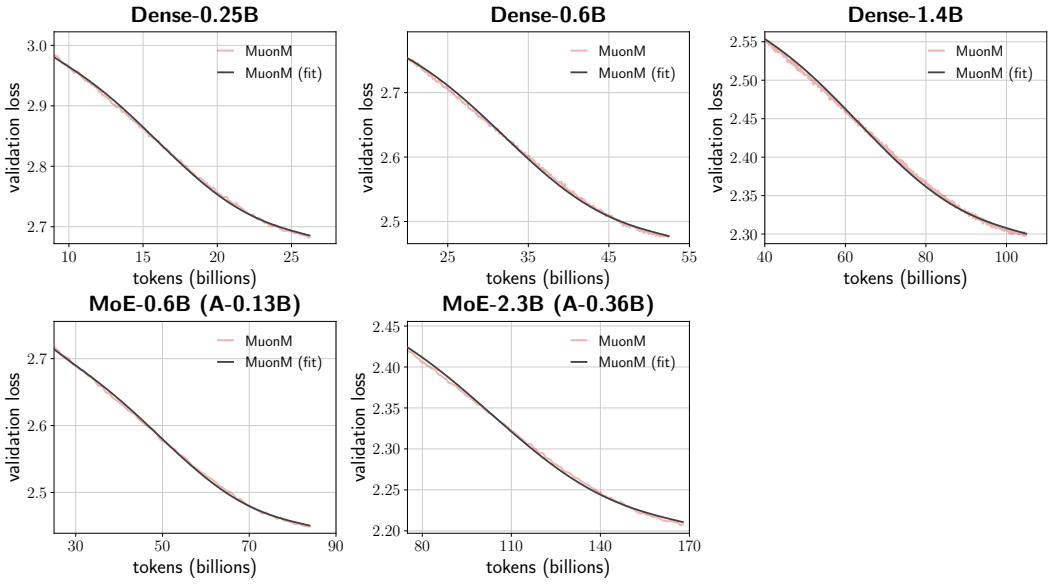

Figure 16: Comparison of fitted and true loss trajectories.

I PROOF

I.1 PROOF OF THE MAIN THEOREM

Notations. Define $\rho_{\text{fast}} = 1 - \alpha_{\text{fast}}, \rho_{\text{slow}} = 1 - \alpha_{\text{slow}}$, $\eta_{\max,k} = \max_{i<k} \eta_i$. For sharp/flat subspace indicator $q \in \{0, 1\}$, write $d_q = \frac{\rho_{\text{fast}}}{\alpha^2_{\text{fast}}} + \chi q \frac{\rho_{\text{slow}}}{\alpha^2_{\text{slow}}}$, $a_0 = a_{\text{top}}$ and $a_1 = a_{\text{full}}$. We take $e = 2.71828...$ as the base of the natural logarithm, unless explicitly defined otherwise.

We first introduce explicit constant coefficients in Assumption 7.1 to facilitate cleaner mathematical derivations.

Assumption I.1 (Detailed Version of Assumption 7.1)**.** There exist constants $\ell_-, \ell_+, \vartheta_-, \vartheta_+ > 0$, a capacity exponent $\nu > 1$ and a source exponent $s > 0$ such that, for every $1 \le j \le d$,

$$\ell_- j^{-\nu} \le \lambda_j \le \ell_+ j^{-\nu}, \qquad \vartheta_- j^{-1} \lambda_j^{s-1} \le |w_j^\star|^2 \le \vartheta_+ j^{-1} \lambda_j^{s-1}.$$

Lemma I.2 (Hessian-aligned anisotropic noise structure)**.** *For every $u \in \mathbb{R}^d$, the gradient noise covariance is*

$$\Sigma(u) = Huu^\top H + (u^\top Hu + \sigma^2)H, \tag{55}$$

and it satisfies

$$(2\mathcal{E}(u) + \sigma^2)H \preceq \Sigma(u) \preceq (4\mathcal{E}(u) + \sigma^2)H. \tag{56}$$

Proof. This lemma is Lemma A.1 in (Wang et al., 2026a), and we restate its proof here for completeness. For one sample, the stochastic gradient satisfies

$$g = \phi\phi^\top u - \phi\epsilon, \qquad \mathbb{E}g = Hu.$$

Wick's formula for the centered Gaussian vector ϕ gives

$$\mathbb{E}[\phi\phi^\top uu^\top \phi\phi^\top] = 2Huu^\top H + (u^\top Hu)H.$$

Further computation based on it proves (55). Finally, we have $Huu^\top H \preceq (u^\top Hu)H$ by Cauchy–Schwarz to the H-inner product. Since $u^\top Hu = 2\mathcal{E}(u)$, this proves (56). □

Remark I.3. Lemma I.2 shows the alignment between the noise and the Hessian. In the early phase of training, $\mathcal{E}$ is large and the noise is dominated by the loss scale. In the later phase, where $\mathcal{E} \lesssim \sigma^2$, the noise aligns well with the Hessian, and the stochastic gradient is noise-dominated.

The following lemma gives the aggregate parameter-state update rule.

Lemma I.4 (Discrete parameter-momentum recursion)**.** *For the j-th eigen-direction $1 \le j \le d$, define*

$$X_{j,k} := (u_{j,k}, m^{\text{fast}}_{j,k-1}, m^{\text{slow}}_{j,k-1})^\top.$$

Then the concrete algorithm (7) *is equivalent with*

$$X_{j,k+1} = M_{j,k} X_{j,k} + d_{j,k}\xi_{j,k}, \tag{57}$$

where

$$M_{j,k} = \begin{pmatrix} 1-\eta_k c_j \lambda_j & -\eta_k \rho_{\text{fast}} & -\eta_k \chi q_j \rho_{\text{slow}} \\ \lambda_j & \rho_{\text{fast}} & 0 \\ \lambda_j & 0 & \rho_{\text{slow}} \end{pmatrix}, \qquad d_{j,k} = \begin{pmatrix} -\eta_k c_j \\ 1 \\ 1 \end{pmatrix}, \tag{58}$$

$c_j := \gamma + 1 + \chi q_j$, and $\xi_{j,k} \in \mathbb{R}$ denotes the shared noise in all three entries.

Proof. Substituting

$$m^{\text{fast}}_k = \rho_{\text{fast}} m^{\text{fast}}_{k-1} + \lambda_j u_{j,k} + \xi_{j,k}, \qquad m^{\text{slow}}_k = \rho_{\text{slow}} m^{\text{slow}}_{k-1} + \lambda_j u_{j,k} + \xi_{j,k}$$

into (7) gives the results. □

We next analyze the roots of the characteristic polynomial $\det(zI - M_{j,k})$ of the transition matrix $M_{j,k}$, which determine the final convergence rate. Consider any fixed coordinate j, and hide the subscript j for brevity. Define the constants

$$a := \gamma + \alpha^{-1}_{\text{fast}} + \widetilde{\chi}\alpha^{-1}_{\text{slow}}, \qquad d := \frac{1-\alpha_{\text{fast}}}{\alpha^2_{\text{fast}}} + \widetilde{\chi}\frac{1-\alpha_{\text{slow}}}{\alpha^2_{\text{slow}}},$$

where $\widetilde{\chi} \ge 0$ denotes the corresponding coefficient for the slow momentum. The matrix with zero learning rate has eigenvalues

$$1, \qquad \rho_{\text{fast}} = 1 - \alpha_{\text{fast}}, \qquad \rho_{\text{slow}} = 1 - \alpha_{\text{slow}}.$$

The next lemma shows that for a sufficiently small positive learning rate, the root issuing from 1 remains separated from the two roots issuing from ρ_{fast} and ρ_{slow}.

Lemma I.5 (Spectrum distribution of the transition matrix)**.** *Let $M_{\widetilde{\chi},\lambda,\eta}$ be the matrix obtained from* (58) *with $\lambda_j = \lambda$, $\chi q_j = \widetilde{\chi}$ and $\eta_k = \eta$. Define $x = \eta\lambda$. If*

$$a\eta\lambda \le \frac{\alpha_{\text{slow}}}{8}, \tag{59}$$

then we have:

- *$M_{\widetilde{\chi},\lambda,\eta}$ has a real simple eigenvalue z_0 in $\left[1 - \frac{\alpha_{\text{slow}}}{4}, 1\right]$ with*

$$\left|\log z_0 + ax + a\left(d + \frac{a}{2}\right)x^2\right| \le c_{\log} x^3, \tag{60}$$

  where $c_{\log} = 4a(d+a)^2 + 6a^2 e$.

- *The other two eigenvalues, counted with algebraic multiplicity, satisfy*

$$\max_{z \in \sigma(M_{\widetilde{\chi},\lambda,\eta}) \setminus \{z_0\}} |z| \le 1 - \frac{3\alpha_{\text{slow}}}{4}. \tag{61}$$

Proof. Expanding $\det(zI - M_{\widetilde{\chi},\lambda,\eta})$ along its first row gives

$$P(z) = (z-\rho_{\text{fast}})(z-\rho_{\text{slow}})(z-1+\eta c\lambda) + \eta\lambda\rho_{\text{fast}}(z-\rho_{\text{slow}}) + \eta\widetilde{\chi}\lambda\rho_{\text{slow}}(z-\rho_{\text{fast}}), \tag{62}$$

where $c = \gamma + 1 + \widetilde{\chi}$. Define the constants $e = 2\frac{1-\alpha_{\text{fast}}}{\alpha^3_{\text{fast}}} + 2\widetilde{\chi}\frac{1-\alpha_{\text{slow}}}{\alpha^3_{\text{slow}}}$, and $c_{\log} = 4a(d+a)^2 + 6a^2 e$.

- Bound the largest real root. Note that $F(z) > 0$ for $z \geq 1$. Thus all real roots are less than 1. Let $\delta := 1 - z_1$. The characteristic equation gives

$$\delta = \underbrace{x\left[c + \frac{\rho_{\text{fast}}}{\alpha_{\text{fast}} - \delta} + \widetilde{\chi}\frac{\rho_{\text{slow}}}{\alpha_{\text{slow}} - \delta}\right]}_{:=F_x(\delta)}. \tag{63}$$

On the interval $I = [0, \alpha_{\text{slow}}/2]$, we have $0 \leq F_x(\delta) \leq 2ax \leq \alpha_{\text{slow}}/4$. Thus $F_x(I) \subset I$. On the other hand, for $\delta \in I$,

$$F_x'(\delta) = x\left[\frac{\rho_{\text{fast}}}{(\alpha_{\text{fast}} - \delta)^2} + \widetilde{\chi}\frac{\rho_{\text{slow}}}{(\alpha_{\text{slow}} - \delta)^2}\right] \leq 4xd \leq \frac{1}{2}.$$

Since F_x is a contraction mapping on I, it admits a unique real fixed point $\delta_x \in I$, with

$$0 \leq \delta_x = F_x(\delta_x) \leq 2ax \leq \frac{\alpha_{\text{slow}}}{4}, \qquad \lim_{x\to 0} \delta_x = 0. \tag{64}$$

Combining it with $P(-\infty) < 0, P(\rho_{\text{slow}}) > 0$, P has at least 2 real roots. Thus P has 3 real roots.

Next we consider finer estimate to the largest root. Expanding the two $(1-\delta)^{-1}$ terms in (63) via their geometric series (with remainder) in powers of δ gives

$$a + d\delta + \frac{e}{2}\delta^2 \leq c + \frac{\rho_{\text{fast}}}{\alpha_{\text{fast}} - \delta} + \widetilde{\chi}\frac{\rho_{\text{slow}}}{\alpha_{\text{slow}} - \delta} \leq a + d\delta + \frac{2e}{3}\delta^2$$

by using $\delta \leq 2ax \leq \alpha_{\text{slow}}/4$. Hence

$$0 \leq \delta - ax \leq 2adx^2 + \frac{8}{3}a^2ex^3. \tag{65}$$

Further applying $dx \leq ax/\alpha_{\text{slow}} \leq 1/8$ gives

$$0 \leq \delta - ax - adx^2 \leq \left(2ad^2 + 3a^2e\right)x^3.$$

Combining them with $\sum_{n\geq 3} \delta^n/n \leq \delta^3/[3(1-\delta)]$ gives

$$\begin{aligned}|\log(1-\delta) + ax + a(d + a/2)x^2| \leq& |\delta - ax - adx^2| + \frac{|\delta^2 - a^2x^2|}{2} + \sum_{n\geq 3}\frac{\delta^n}{n} \\ \leq& \left(2ad^2 + 3a^2e\right)x^3 + \left(3a^2d + \frac{1}{2}a^2e\right)x^3 + 4a^3x^3.\end{aligned} \tag{66}$$

Their sum is at most $c_{\log}x^3$, proving (60).

- Bound the remaining two roots. Define

$$T = \begin{pmatrix} 1 & 0 & 0 \\ -1 & \alpha_{\text{fast}}/\lambda & 0 \\ -1 & 0 & \alpha_{\text{slow}}/\lambda \end{pmatrix},$$

and $b_{\text{fast}} := \frac{\rho_{\text{fast}}}{\alpha_{\text{fast}}}$, $b_{\text{slow}} := \widetilde{\chi}\frac{\rho_{\text{slow}}}{\alpha_{\text{slow}}}$. Then M is similar to

$$N_q(x) := TMT^{-1} = \begin{pmatrix} 1 - ax & -xb_{\text{fast}} & -xb_{\text{slow}} \\ ax & \rho_{\text{fast}} + xb_{\text{fast}} & xb_{\text{slow}} \\ ax & xb_{\text{fast}} & \rho_{\text{slow}} + xb_{\text{slow}} \end{pmatrix}. \tag{67}$$

Thus $\|N_q(x) - \text{diag}(1, \rho_{\text{fast}}, \rho_{\text{slow}})\|_\infty \leq 2ax \leq \alpha_{\text{slow}}/4$. The Gershgorin disk associated with the first row lies in $\text{Re}\, z \geq 1 - 2ax \geq 1 - \alpha_{\text{slow}}/4$. Each of the other two disks lies in $|z| \leq \rho_{\text{slow}} + 2ax \leq 1 - 3\alpha_{\text{slow}}/4$. The first disk is therefore disjoint from the union of the other two. By the Gershgorin disk theorem (Theorem I.17), the first component contains exactly one eigenvalue, while the complementary component contains exactly two eigenvalues counted with algebraic multiplicity, proving (61).

□

Lemma I.6 (Properties of spectral projections). *We use the notation $N_q(x)$ in* (67)*, where $q \in \{0,1\}$ is the sharp subspace indicator. For $x \geq 0$ satisfying $a_q x \leq \alpha_{\text{slow}}/8$, let*

$$\Gamma_0 := \{z \in \mathbb{C} : |z-1| = \alpha_{\text{slow}}/2\}, \qquad \Gamma_1 := \{z \in \mathbb{C} : |z| = 1 - \alpha_{\text{slow}}/2\},$$
$$P_0(x) := \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_0} (zI - N_q(x))^{-1}\, \mathrm{d}z, \quad P_1(x) := \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_1} (zI - N_q(x))^{-1}\, \mathrm{d}z. \tag{68}$$

Both contours are positively oriented. The following properties hold.

(i) The operators in (68) *are complementary oblique subspace projections with $P_0(x) + P_1(x) = I$ and $P_0(x)P_1(x) = P_1(x)P_0(x) = 0$. $P_0(x)$ is associated with the largest eigenvalue and $P_1(x)$ is associated with the complementary two-dimensional subspace*[4]*. If $x, y \geq 0$ and $a_q \max\{x, y\} \leq \alpha_{\text{slow}}/8$, then*

$$\|P_0(x)\|_\infty \leq 2, \qquad \|P_0(x) - P_0(y)\|_\infty \leq \frac{16 a_q}{\alpha_{\text{slow}}} |x - y|, \tag{69}$$

and

$$\max\{\|P_1(y)P_0(x)\|_\infty, \|P_0(y)P_1(x)\|_\infty\} \leq \frac{32 a_q}{\alpha_{\text{slow}}} |x - y|. \tag{70}$$

(ii) At $x = 0$, one has $P_0(0) = \operatorname{diag}(1, 0, 0)$. Besides, the projector P_0 is differentiable at $x = 0$ with

$$P_0'(0) = \begin{pmatrix} 0 & -\rho_{\text{fast}}/\alpha_{\text{fast}}^2 & -\chi q \rho_{\text{slow}}/\alpha_{\text{slow}}^2 \\ a_q/\alpha_{\text{fast}} & 0 & 0 \\ a_q/\alpha_{\text{slow}} & 0 & 0 \end{pmatrix}, \tag{71}$$

and its quadratic remainder satisfies

$$\|P_0(x) - P_0(0) - x P_0'(0)\|_\infty \leq \frac{32 a_q^2 x^2}{\alpha_{\text{slow}}^2}. \tag{72}$$

(iii) Let $\delta = 1 - z_{0,q}(x)$. The range of $P_0(x)$ is spanned by the eigenvector

$$r_q(x) = \left(1, \frac{\delta}{\alpha_{\text{fast}} - \delta}, \frac{\delta}{\alpha_{\text{slow}} - \delta}\right)^\top \text{ with } \|r_q(x)\|_\infty = 1. \tag{73}$$

The complementary projection P_1 satisfies for any $v \in \operatorname{Range} P_1(x)$,

$$\|N_q(x) v\|_\infty \leq \left(1 - \frac{17 \alpha_{\text{slow}}}{20}\right) \|v\|_\infty. \tag{74}$$

(iv) Let $a \leq b$, $x_m \geq 0$, and suppose $a_q x_m \leq \alpha_{\text{slow}}/8$ for $a \leq m \leq b$. Define

$$L_m := \|P_0(x_m) - P_0(x_{m-1})\|_\infty.$$

With the ordered-product notation $\overleftarrow{\prod_{m=a}^{b}} A_m = A_b \cdots A_a$, we have

$$\left\| \overleftarrow{\prod_{m=a}^{b}} \big[N_q(x_m) P_1(x_m)\big] v \right\|_\infty \leq e^{\sum_{m=a+1}^{b} L_m} \left(1 - \frac{17 \alpha_{\text{slow}}}{20}\right)^{b-a+1} \|P_1(x_a) v\|_\infty. \tag{75}$$

Proof. Proof of (i). Lemma I.5 shows that Γ_0 encloses exactly the largest root $z_{0,q}(x)$ (i.e, the largest eigenvalue of $N_q(x)$). The disk bounded by Γ_1 contains the other two eigenvalues, counted with algebraic multiplicity, and excludes $z_{0,q}(x)$. Hence P_0 is the (rank 1) oblique projection to the eigenspace corresponding to $z_{0,q}(x)$, and $P_1 = I - P_0$. On each contour, the Neumann resolvent identity and $\|N_q(x) - N_q(0)\|_\infty \leq 2a_q x$ give

$$\|(zI - N_q(x))^{-1}\|_\infty \leq \frac{4}{\alpha_{\text{slow}}}.$$

[4] In general, P_0, P_1 are not orthogonal to each other.

The Riesz formula and the second resolvent identity now give (69). Complementarity gives

$$P_1(y)P_0(x) = [P_0(x) - P_0(y)]P_0(x), \qquad P_0(y)P_1(x) = P_0(y)[P_0(y) - P_0(x)],$$

so (70) follows from (69).

Proof of (ii). Write $N_q(x) = N_q(0) + xN_q'(0)$ (which is linear). Differentiating the Riesz formula at $x = 0$ gives

$$P_0'(0) = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma_0} R_0(z)N_q'(0)R_0(z)\,\mathrm{d}z = \begin{pmatrix} 0 & -\rho_{\mathrm{fast}}/\alpha_{\mathrm{fast}}^2 & -\chi q\rho_{\mathrm{slow}}/\alpha_{\mathrm{slow}}^2 \\ a_q/\alpha_{\mathrm{fast}} & 0 & 0 \\ a_q/\alpha_{\mathrm{slow}} & 0 & 0 \end{pmatrix}, \quad (76)$$

where $R_x(z) := (zI - N_q(x))^{-1}$ and we use $R_0(z) = \mathrm{diag}((z-1)^{-1}, (z-\rho_{\mathrm{fast}})^{-1}, (z-\rho_{\mathrm{slow}})^{-1})$. Applying the second-order resolvent identity

$$R_x = R_0 + xR_0N_q'(0)R_0 + x^2R_0N_q'(0)R_xN_q'(0)R_0,$$

we have

$$\begin{aligned} P_0(x) - P_0(0) - xP_0'(0) =& \frac{1}{2\pi i} \int_{\Gamma_0} \left(R_x(z) - R_0(z) - xR_0(z)N_q'(0)R_0(z)x\right)\, dz \\ =& \frac{x^2}{2\pi i} \int_{\Gamma_0} \left(R_0(z)N_q'(0)R_x(z)N_q'(0)R_0(z)\right)\, dz \end{aligned} \quad (77)$$

Using the bounds $\|R_0\|_\infty \le \frac{2}{\alpha_{\mathrm{slow}}}$, $\|R_x\|_\infty \le \frac{4}{\alpha_{\mathrm{slow}}}$, $\|N_q'(0)\|_\infty \le 2a_q$, we get

$$\begin{aligned} \|P_0(x) - P_0(0) - xP_0'(0)\|_\infty \le& \frac{x^2}{2\pi}|\Gamma_0|\|R_0\|_\infty^2\|N_q'(0)\|_\infty^2\|R_x\|_\infty \\ \le& \frac{x^2}{2\pi}(\pi\alpha_{\mathrm{slow}}) \left(\frac{2}{\alpha_{\mathrm{slow}}}\right)^2 (2a_q)^2 \frac{4}{\alpha_{\mathrm{slow}}} \le \frac{32a_q^2x^2}{\alpha_{\mathrm{slow}}^2}. \end{aligned} \quad (78)$$

Proof of (iii). Direct computation verifies that $N_q(x)r_q(x) = z_{0,q}(x)r_q(x)$. Since $0 \le \delta \le \alpha_{\mathrm{slow}}/4 < \alpha_{\mathrm{fast}}$, the last two coordinates of $r_q(x)$ belong to $[0, 1/3]$, proving (73).

For the complementary subspace, set $\theta_q(x) := x\left(\frac{\rho_{\mathrm{fast}}}{\alpha_{\mathrm{fast}}(\alpha_{\mathrm{fast}}-\delta)} + \frac{\chi q\rho_{\mathrm{slow}}}{\alpha_{\mathrm{slow}}(\alpha_{\mathrm{slow}}-\delta)}\right)$. Then by computation, an eigenvector for $z_{0,q}(x)$ with respect to $N_q(x)^\top$ is

$$\ell_q(x) = \left(1, -\frac{x\rho_{\mathrm{fast}}}{\alpha_{\mathrm{fast}}(\alpha_{\mathrm{fast}} - \delta)(1 - \theta_q(x))}, -\frac{x\chi q\rho_{\mathrm{slow}}}{\alpha_{\mathrm{slow}}(\alpha_{\mathrm{slow}} - \delta)(1 - \theta_q(x))}\right)^\top.$$

The bounds $\delta \le \alpha_{\mathrm{slow}}/4$, $d_q \le a_q/\alpha_{\mathrm{slow}}$, and $a_qx \le \alpha_{\mathrm{slow}}/8$ imply

$$0 \le \theta_q(x) \le \frac{4}{3}xd_q \le \frac{1}{6}.$$

Because $\mathrm{Range}\, P_1(x) = \ker \ell_q(x)^\top$, every vector in the fast range satisfies

$$|v_1| \le \frac{\theta_q(x)}{1 - \theta_q(x)} \max\{|v_2|, |v_3|\} \le \frac{1}{5} \max\{|v_2|, |v_3|\}.$$

The last two coordinates of $N_q(x)v$ are therefore bounded by

$$\left(1 - \alpha_{\mathrm{slow}} + \frac{6}{5}a_qx\right) \|v\|_\infty \le \left(1 - \frac{17\alpha_{\mathrm{slow}}}{20}\right) \|v\|_\infty.$$

This proves (74).

Proof of (iv). If $v \in \mathrm{Range}\, P_1(x_{m-1})$, then $\|P_1(x_m)v\|_\infty \le (1 + L_m)\|v\|_\infty$. Iteration with (74) proves (75). □

Before proceeding with the proof, we recall some previous notations and introduce a few new ones.

$$a_j := \begin{cases} a_{\mathrm{top}}, & j \le r, \\ a_{\mathrm{full}}, & j > r, \end{cases} \qquad A := a_{\mathrm{top}}(I_d - Q) + a_{\mathrm{full}} Q,$$
$$S_A(t) := \sum_{j=1}^{d} \lambda_j |w_j^\star|^2 e^{-2a_j\lambda_j t}, \qquad K_A(t) := \sum_{j=1}^{d} a_j^2 \lambda_j^2 e^{-2a_j\lambda_j t}. \tag{79}$$

Lemma I.7 (Uniform kernel bounds)**.** *Suppose that Assumption I.1 and $d \ge 2(r+1)$ hold. Then:*

(i) There are positive constants $c_{S,-}, c_{S,+}, c_{K,-}, c_{K,+}$, given in the proof and depending only on the power-law constants, s, ν, and C_d, such that, for $0 \le a_{\mathrm{full}} t \le C_d d^\nu$,

$$\min\{1, c_{S,-}\}\mathsf{S}(t) \le S_A(t) \le \max\{1, c_{S,+}\}\mathsf{S}(t),$$
$$\min\{1, c_{K,-}\}\mathsf{K}(t) \le K_A(t) \le \max\{1, c_{K,+}\}\mathsf{K}(t). \tag{80}$$

(ii) The kernel K_A satisfies

$$\int_0^\infty K_A(t)\,\mathrm{d}t = \frac{1}{2}\operatorname{tr}(AH). \tag{81}$$

(iii) If $(r+1)^\nu \le a_{\mathrm{full}} t \le C_d d^\nu$, then

$$\mathsf{S}(t) \asymp \sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\mathrm{top}}\lambda_j t} + (a_{\mathrm{full}} t)^{-s}, \qquad \mathsf{K}(t) \asymp a_{\mathrm{full}}^{1/\nu} t^{-p}. \tag{82}$$

(iv) There is a constant C_h depending only on the power-law constants, s, and ν, such that, if $r \ge 4$ and $a_{\mathrm{top}} t \ge C_h r^\nu (1 + \log\kappa)$, then

$$\sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\mathrm{top}}\lambda_j t} \lesssim (a_{\mathrm{full}} t)^{-s}, \tag{83}$$

where $\kappa := a_{\mathrm{full}}/a_{\mathrm{top}} \ge 1$.

Proof. Throughout the proof, we abbreviate $\beta_S := 1 + s\nu$, $\beta_K := 2\nu$, $R := r + 1$.

Proof of (i). For $\beta > 1$, define

$$L_1(\beta) := \min\left\{ 4^{-\beta} e^{-2\ell_+}, \frac{1}{2}\, 4^{1-\beta} e^{-2^{\nu+1}\ell_+ C_d} \right\},$$

and set

$$c_{S,-} := \vartheta_- \ell_-^s L_1(\beta_S), \tag{84}$$
$$c_{S,+} := \vartheta_+ \ell_+^s 2^s \max\left\{ 1 + \frac{1}{s\nu}, \frac{\Gamma(s)}{\nu(2\ell_-)^s} + 2\left(\frac{s+\nu^{-1}}{2e\ell_-}\right)^{s+\nu^{-1}} \right\}, \tag{85}$$
$$c_{K,-} := \ell_-^2 L_1(\beta_K), \tag{86}$$
$$c_{K,+} := \ell_+^2 2^p \max\left\{ 1 + \frac{1}{2\nu - 1}, \frac{\Gamma(p)}{\nu(2\ell_-)^p} + \frac{8e^{-2}}{(2\ell_-)^2} \right\}, \tag{87}$$

where $\Gamma(a) := \int_0^\infty y^{a-1} e^{-y}\, dy$ denotes the Gamma function for $a > 0$. The following proof uses that $c_- A \le D \le c_+ A$ implies $\min\{1, c_-\}(B + A) \le B + D \le \max\{1, c_+\}(B + A)$ for positive A, B, D.

- *Lower bounds.* Set $u = a_{\mathrm{full}} t$, and for any $\beta > 1$,

$$F_{\beta,d}(u) := \sum_{j=R}^{d} j^{-\beta} e^{-2\ell_+ u j^{-\nu}}, \qquad \delta_\beta := \frac{\beta - 1}{\nu}.$$

We prove that the tail estimate

$$F_{\beta,d}(u) \ge L_1(\beta)(R^\nu + u)^{-\delta_\beta} \tag{88}$$

holds for any $0 \le u \le C_d d^\nu$.

– *Case 1:* $u \le R^\nu$. We have

$$F_{\beta,d}(u) \ge \sum_{j=R}^{2R-1} j^{-\beta} e^{-2\ell_+ u j^{-\nu}} \ge 2^{-\beta} e^{-2\ell_+} R^{1-\beta} \ge 4^{-\beta} e^{-2\ell_+} (R^\nu + u)^{-\delta_\beta}.$$

– *Case 2:* $u > R^\nu$. Define $J := u^{1/\nu} > R$.
If $J \le d/4$,

$$F_{\beta,d}(u) \ge \sum_{J \le j \le 4J} j^{-\beta} e^{-2\ell_+ u j^{-\nu}} \ge 4^{-\beta} e^{-2\ell_+} J^{1-\beta}.$$

If $J > d/4$, $u \le C_d d^\nu$ gives $u j^{-\nu} \le u(d/2)^{-\nu} \le C_d d^\nu (d/2)^{-\nu} = 2^\nu C_d$. It follows that

$$F_{\beta,d}(u) \ge \sum_{\frac{d}{2} \le j \le d} j^{-\beta} e^{-2\ell_+ u j^{-\nu}} \ge \frac{d}{2}\, d^{-\beta} e^{-2^{\nu+1}\ell_+ C_d} \ge \frac{1}{2}\, 4^{1-\beta} e^{-2^{\nu+1}\ell_+ C_d} J^{1-\beta},$$

where the last inequality uses $d < 4J$.
In both cases, $(R^\nu + u)^{-\delta_\beta} \le u^{-\delta_\beta} = J^{1-\beta}$, proving (88).

For the signal lower bound, since $\lambda_j |w_j^\star|^2 e^{-2a_{\mathrm{full}}\lambda_j t} \ge \vartheta_- \ell_-^s j^{-\beta_S} e^{-2\ell_+ u j^{-\nu}}$,

$$\sum_{j=R}^{d} \lambda_j |w_j^\star|^2 e^{-2a_{\mathrm{full}}\lambda_j t} \ge \vartheta_- \ell_-^s F_{\beta_S,d}(u) \ge \vartheta_- \ell_-^s L_1(\beta_S)(R^\nu + u)^{-(\beta_S - 1)/\nu} \ge c_{S,-}(R^\nu + u)^{-s}.$$

For the kernel lower bound, we use $a_{\mathrm{full}}^2 \lambda_j^2 e^{-2a_{\mathrm{full}}\lambda_j t} \ge a_{\mathrm{full}}^2 \ell_-^2 j^{-2\nu} e^{-2\ell_+ u j^{-\nu}}$ and get

$$\sum_{j=R}^{d} a_{\mathrm{full}}^2 \lambda_j^2 e^{-2a_{\mathrm{full}}\lambda_j t} \ge a_{\mathrm{full}}^2 \ell_-^2 F_{\beta_K,d}(u) \ge a_{\mathrm{full}}^2 \ell_-^2 L_1(\beta_K)(R^\nu + u)^{-(\beta_K - 1)/\nu} \ge c_{K,-} a_{\mathrm{full}}^2 (R^\nu + u)^{-p}.$$

- *Upper bounds.* For the upper bounds, it suffices to control the sum

$$G_\beta(u) := \sum_{j=R}^{\infty} j^{-\beta} e^{-2\ell_- u j^{-\nu}}, \qquad \beta > 1.$$

– *Case 1:* $u \le R^\nu$. We have

$$G_\beta(u) \le \sum_{j=R}^{\infty} j^{-\beta} \le R^{1-\beta} \left(1 + \frac{1}{\beta - 1}\right).$$

– *Case 2:* $u > R^\nu$. Define

$$f_{\beta,u}(x) := x^{-\beta} e^{-2\ell_- u x^{-\nu}}.$$

We use the elementary inequality

$$\sum_{j \ge 1} f(j) \le \int_0^\infty f(x)\, dx + 2 \sup_{x>0} f(x).$$

For the integral term, set $\delta_\beta = \frac{\beta - 1}{\nu}$ and make the change of variables $y = 2\ell_- u x^{-\nu}$. This yields

$$\int_0^\infty f(x)\, dx = \int_0^\infty x^{-\beta} e^{-2\ell_- u x^{-\nu}}\, dx = \frac{(2\ell_- u)^{-\delta_\beta}}{\nu} \int_0^\infty y^{\delta_\beta - 1} e^{-y}\, dy = \frac{\Gamma(\delta_\beta)}{\nu (2\ell_- u)^{\delta_\beta}}.$$

To bound the supremum term, we compute the derivative $\frac{d}{dx} \log f_{\beta,u}(x) = -\frac{\beta}{x} + 2\ell_- u \nu x^{-\nu-1}$. Hence the maximum is attained at the point satisfying $2\ell_- u \nu x^{-\nu} = \beta$. Thus $\sup_{x>0} f_{\beta,u}(x) = \left(\frac{\beta/\nu}{2e\ell_- u}\right)^{\beta/\nu}$. Since $u > R^\nu \ge 1$ and $\beta/\nu = \delta_\beta + \nu^{-1} > \delta_\beta$, it follows that

$$G_\beta(u) \le \left[\frac{\Gamma(\delta_\beta)}{\nu (2\ell_-)^{\delta_\beta}} + 2\left(\frac{\beta/\nu}{2e\ell_-}\right)^{\beta/\nu}\right] u^{-\delta_\beta}.$$

In either case, $R^\nu + u \le 2\max\{R^\nu, u\}$. Consequently,

$$G_\beta(u) \le 2^{\delta_\beta} \max\left\{1 + \frac{1}{\beta - 1}, \frac{\Gamma(\delta_\beta)}{\nu(2\ell_-)^{\delta_\beta}} + 2\left(\frac{\beta/\nu}{2e\ell_-}\right)^{\beta/\nu}\right\}(R^\nu + u)^{-\delta_\beta}. \tag{89}$$

Substituting the value of β into it yields the upper bounds in (80).

Proof of (ii). Direct computation gives

$$\int_0^\infty K_A(t)\,\mathrm{d}t = \sum_{j=1}^d a_j^2\lambda_j^2 \frac{1}{2a_j\lambda_j} = \frac{1}{2}\sum_{j=1}^d a_j\lambda_j = \frac{1}{2}\operatorname{tr}(AH).$$

Proof of (iii). Put $u = a_{\text{full}}t$. Since $y^2e^{-2y} \le e^{-2}$ for $y \ge 0$, and $r \le u^{1/\nu}$, we have

$$\sum_{j=1}^r a_{\text{top}}^2\lambda_j^2 e^{-2a_{\text{top}}\lambda_j t} = a_{\text{full}}^2 u^{-2}\sum_{j=1}^r (\kappa^{-1}\lambda_j u)^2 e^{-2b\lambda_j u} \le e^{-2}a_{\text{full}}^2 u^{-2} r \le e^{-2}a_{\text{full}}^2 u^{-p}.$$

This proves the forgetting kernel part in (82). The signal rate part is obtained by an analogous argument, where the sharp subspace summation is retained.

Proof of (iv). Define the constants $c = 2\ell_-, x = a_{\text{top}}t, y = \frac{x}{r^\nu}$ for notational brevity. We suppose $y \ge 1$ and will verify this condition later. For $\beta > 1$, set the constant

$$C_\beta^{top} = 2^\beta\left[\max\left\{1, \frac{1}{c\nu}\right\} + \frac{2}{ec(2^\nu - 1)}\sum_{m\ge 1} 2^{m(\beta-1)}e^{-\frac{c}{2}(2^{m\nu}-1)}\right]. \tag{90}$$

We first consider the sum on $r/2 \le j \le r$. With $n = r - j$, Bernoulli's inequality gives $\left(\frac{r}{j}\right)^\nu = \left(1 - \frac{n}{r}\right)^{-\nu} \ge 1 + \frac{\nu n}{r}$. Since $j^{-\beta} \le 2^\beta r^{-\beta}$, we have

$$\sum_{r/2\le j\le r} j^{-\beta}e^{-cy(r/j)^\nu} \le 2^\beta r^{-\beta}e^{-cy}\sum_{n\ge 0} e^{-c\nu y n/r} \le 2^\beta \max\left\{1, \frac{1}{c\nu}\right\} r^{1-\beta}(y^{-1} + r^{-1})e^{-cy}, \tag{91}$$

where the last inequality uses $(1 - e^{-z})^{-1} \le 1 + z^{-1}$ for $z > 0$.

For the remaining summation, we define

$$B_m := \left\{j : \frac{r}{2^{m+1}} < j \le \frac{r}{2^m}\right\}, \qquad m \ge 1.$$

On B_m,

$$j^{-\beta} \le \left(\frac{2^{m+1}}{r}\right)^\beta, \qquad \left(\frac{r}{j}\right)^\nu \ge 2^{m\nu}.$$

Consequently,

$$\sum_{j\in B_m} j^{-\beta}e^{-cy(r/j)^\nu} \le 2^\beta r^{1-\beta}2^{m(\beta-1)}e^{-cy2^{m\nu}} = 2^\beta r^{1-\beta}y^{-1}e^{-cy}2^{m(\beta-1)}\left[ye^{-c(2^{m\nu}-1)y}\right]. \tag{92}$$

For $a_m := c(2^{m\nu} - 1)$, one has

$$ye^{-a_m y} = \left(ye^{-a_m y/2}\right)e^{-a_m y/2} \le \frac{2}{ea_m}e^{-a_m/2} \le \frac{2}{ec(2^\nu - 1)}e^{-a_m/2}.$$

Thus

$$\sum_{j\le r/2} j^{-\beta}e^{-cy(r/j)^\nu} \le 2^\beta\frac{2}{ec(2^\nu - 1)}r^{1-\beta}y^{-1}e^{-cy}\sum_{m\ge 1} 2^{m(\beta-1)}e^{-\frac{c}{2}(2^{m\nu}-1)}.$$

Combining it with (91) gives

$$\sum_{j=1}^{r} j^{-\beta} e^{-cxj^{-\nu}} \le C_\beta^{top} r^{1-\beta}(y^{-1}+r^{-1})e^{-cy}. \tag{93}$$

Applying (93) with $\beta = \beta_S = 1 + s\nu$ gives

$$\sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\text{top}}\lambda_j t} \le \vartheta_+ \ell_+^s \sum_{j=1}^{r} j^{-\beta_S} e^{-cxj^{-\nu}} \le C_S^{top} r^{-s\nu}(y^{-1}+r^{-1})e^{-2\ell_- y}, \tag{94}$$

where the constant $C_S^{top} := \vartheta_+ \ell_+^s C_{\beta_S}^{top}$ depends only on the power-law constants, s, and ν.

Finally, set

$$C_h := \max\left\{1, \frac{s}{\ell_-}\right\}.$$

Then $y \ge C_h(1+\log\kappa) \ge 1$. Since $\kappa x = a_{\text{full}} t$, division by $(\kappa x)^{-s} = \kappa^{-s} r^{-s\nu} y^{-s}$ gives

$$\frac{\sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\text{top}}\lambda_j t}}{(a_{\text{full}} t)^{-s}} \le C_S^{top}\big(y^{s-1} + r^{-1}y^s\big)e^{-\ell_- y} \le C_S^{top} \sup_{z\ge 1}(z^{s-1}+z^s)e^{-\ell_- z} < \infty,$$

where the right hand side is a finite constant. This proves (83).

□

We now consider iterating the parameter-momentum update as specified in Lemma I.4. For the j-th coordinate and integers $k > i$, define

$$\Phi_{j;k,i} := M_{j,k-1} M_{j,k-2} \cdots M_{j,i}, \qquad \Phi_{j;i,i} := I_3,$$

and

$$R_{j;k,i} := e_1^\top \Phi_{j;k,i+1} d_{j,i}, \tag{95}$$

where $e_1 = (1,0,0)^\top$. Let

$$e_k^{\text{signal}} := \frac{1}{2}\sum_{j=1}^{d} \lambda_j \left| e_1^\top \Phi_{j;k,0} X_{j,0} \right|^2, \qquad K_{k,i}^{\text{noise}} := \sum_{j=1}^{d} \lambda_j^2 R_{j;k,i}^2. \tag{96}$$

Lemma I.8 (Two-sided convolution-type bounds on $\mathcal{E}$)**.** *For the iterates w_k generated by* (7)*,*

$$e_k^{\text{signal}} + \sum_{i<k} K_{k,i}^{\text{noise}} \left(\mathbb{E}\mathcal{E}(w_i) + \frac{\sigma^2}{2}\right) \le \mathbb{E}\mathcal{E}(w_k) \le e_k^{\text{signal}} + \sum_{i<k} K_{k,i}^{\text{noise}} \left(2\mathbb{E}\mathcal{E}(w_i) + \frac{\sigma^2}{2}\right). \tag{97}$$

Proof. Iterating (57) yields

$$u_{j,k} = e_1^\top \Phi_{j;k,0} X_{j,0} + \sum_{i<k} R_{j;k,i}\xi_{j,i}.$$

Since the noise is independent and has zero mean, it follows that

$$\mathbb{E}\mathcal{E}(w_k) = \frac{1}{2}\sum_{j=1}^{d} \lambda_j u_{j,k}^2 = e_k^{\text{signal}} + \frac{1}{2}\sum_{i<k}\sum_{j=1}^{d} \lambda_j R_{j;k,i}^2\, \mathbb{E}[e_j^\top \Sigma(u_i) e_j].$$

Applying Lemma I.2 proves (97). □

Next we replace $e_k^{\text{signal}}, K_{k,i}^{\text{noise}}$ with quantities related to S_A, K_A, which have closed forms defined in (79). The following lemma characterizes the approximation error of this replacement. We first introduce some definitions for notational brevity. Let

$$V_k(\eta) := \sum_{i=0}^{k-2} |\eta_{i+1} - \eta_i|, \quad \delta_{\text{step}}(k) := \max_{0\le i\le k-2} \frac{|\eta_{i+1}-\eta_i|}{\alpha_{\text{slow}}\eta_i}, \quad \delta_{\text{step}}(1) := 0.$$

Define

$$\varepsilon_{proj}(k) := \frac{a_{\text{full}}\lambda_1}{\alpha_{\text{slow}}^2}\big(\eta_{\max,k} + V_k(\eta)\big) + \frac{\delta_{\text{step}}(k)}{\alpha_{\text{slow}}}, \tag{98}$$

$$\varepsilon_{exp}(k) := a_{\text{full}}\left(d_1 + \frac{a_{\text{full}}}{2}\right)\lambda_1^2\sum_{i<k}\eta_i^2 + c_{\log,1}\lambda_1^3\sum_{i<k}\eta_i^3, \tag{99}$$

$$\varepsilon_{ep}(k) := 64\varepsilon_{proj}(k) + 8\varepsilon_{exp}(k), \tag{100}$$

and for $q \in \{0,1\}$,

$$G_q := \left(\frac{\rho_{\text{fast}}^2}{\alpha_{\text{fast}}^2(1-\rho_{\text{fast}}^2)} + \frac{2\chi q\rho_{\text{fast}}\rho_{\text{slow}}}{\alpha_{\text{fast}}\alpha_{\text{slow}}(1-\rho_{\text{fast}}\rho_{\text{slow}})} + \frac{\chi^2 q^2\rho_{\text{slow}}^2}{\alpha_{\text{slow}}^2(1-\rho_{\text{slow}}^2)}\right)^{\frac{1}{2}}. \tag{101}$$

For any nonnegative sequence $\{s_i\}_i$, define

$$\begin{aligned} L_{j,k} &:= \eta_{k-1} + \sum_{t=0}^{k-2}\prod_{m=t+2}^{k-1}(1 - a_{q_j}\lambda_j\eta_m)\left|\eta_{t+1} - \eta_t + a_{q_j}\lambda_j\eta_t\eta_{t+1}\right|, \\ \widehat{\delta}(k) &:= 4\sum_{j=1}^{d}\lambda_j^2 G_{q_j}^2 L_{j,k}^2, \qquad (\widehat{\mathcal{B}}s)_k := \widehat{\delta}(k)\max_{0\le i<k} s_i. \end{aligned} \tag{102}$$

Lemma I.9 (Approximation errors)**.** *For any nonnegative sequence $\{s_i\}$, define the operator $\mathcal{T}$ that maps s to a sequence $\{\mathcal{T}s\}_i$ as*

$$(\mathcal{T}s)_k := \sum_{i<k}\eta_i^2 K_A(T_k - T_i)s_i, \qquad (\mathcal{T}s)_0 = 0. \tag{103}$$

Suppose

$$a_{\text{full}}\lambda_1\eta_{\max,k} \le \frac{\alpha_{\text{slow}}}{8}, \qquad \varepsilon_{ep}(k) \le \frac{1}{16}. \tag{104}$$

Then

$$\left|e_k^{\text{signal}} - \frac{1}{2}S_A(T_k)\right| \le 4\varepsilon_{ep}(k)S_A(T_k), \tag{105}$$

and

$$\left|\sum_{i=0}^{k-1}K_{k,i}^{\text{noise}}s_i - (\mathcal{T}s)_k\right| \le \big(2\varepsilon_{ep}(k) + \varepsilon_{ep}(k)^2\big)(\mathcal{T}s)_k + 2\big(1+\varepsilon_{ep}(k)\big)\sqrt{(\mathcal{T}s)_k(\widehat{\mathcal{B}}s)_k} + (\widehat{\mathcal{B}}s)_k. \tag{106}$$

Further using Cauchy–Schwarz Inequality to (106) *gives*

$$\frac{225}{512}(\mathcal{T}s)_k - (\widehat{\mathcal{B}}s)_k \le \sum_{i=0}^{k-1}K_{k,i}^{\text{noise}}s_i \le \frac{289}{128}(\mathcal{T}s)_k + 2(\widehat{\mathcal{B}}s)_k. \tag{107}$$

Proof. Fix a coordinate j with eigenvalue $\lambda := \lambda_j$ and sharp subspace indicator $q := q_j \in \{0,1\}$. Set $x_m := \eta_m\lambda$, and consider the projectors from Lemma I.6. The first condition in (104) ensures $a_q x_m \le \alpha_{\text{slow}}/8$ at every step. Suppress q from the projector notation and define

$$L_m := \|P_0(x_m) - P_0(x_{m-1})\|_\infty, \qquad m \ge 1.$$

The estimate in (69) gives

$$\sum_{m=1}^{k-1}L_m \le \frac{16a_q\lambda}{\alpha_{\text{slow}}}V_k(\eta) \le 16\varepsilon_{proj}(k) \le \frac{1}{64}. \tag{108}$$

We considner the matrix $N_q(x)$ in Lemma I.6, where $q \in \{0,1\}$ is the sharp subspace indicator. To be specific, let

$$T_\lambda := \begin{pmatrix} 1 & 0 & 0 \\ -1 & \alpha_{\text{fast}}/\lambda & 0 \\ -1 & 0 & \alpha_{\text{slow}}/\lambda \end{pmatrix}.$$

Then $T_{\lambda_j} M_{j,\ell} T_{\lambda_j}^{-1} = N_q(x_{\ell,j})$ and $T_\lambda e_1 = e_1$, where $x_{\ell,j} := \eta_\ell \lambda_j$. When no ambiguity arises, we omit the subscript j for brevity. We define the ordered matrix product $\overleftarrow{\prod_{\ell=1}^{k}} A_i := A_k A_{k-1} ... A_1$.

For the deterministic part X, since $X_{j,0} = (-w_j^\star, 0, 0)^\top$, the T transformed coordinates is $T_\lambda X_{j,0} = -w_j^\star h$ for $h = (1, -1, -1)^\top$. Define

$$D_{j,k}^s := -w_j^\star e_1^\top \overleftarrow{\prod_{m=0}^{k-1}} \big[N_q(x_m) P_0(x_m)\big] h, \qquad \overline{U}_{j,k} := -w_j^\star \prod_{m=0}^{k-1} z_{0,q}(x_m), \qquad U_{j,k}^0 := -w_j^\star e^{-a_j \lambda_j T_k}.$$

Next we bound each of relevant approximation errors to prove the corresponding conclusions.

Part 1: approximation errors related to $D_{j,k}^s - U_{j,k}^0$.

Recall that $c_q = \gamma + 1 + \chi q$, $d_q = \frac{1-\alpha_{\text{fast}}}{\alpha_{\text{fast}}^2} + \chi q \frac{1-\alpha_{\text{slow}}}{\alpha_{\text{slow}}^2}$. Set

$$p_0 := P_0(x_0) h, \qquad p_m := P_0(x_m) p_{m-1}, \quad 1 \le m < k.$$

Note that $e_1^\top P_0'(0) h = d_q$ by (71) and $P_0(0) h = e_1$. Hence using (72), $d_q \le a_q/\alpha_{\text{slow}}$ and $a_q x_0 \le \alpha_{\text{slow}}/8$, we obtain

$$\begin{aligned} |e_1^\top p_0 - 1| =& |x_0 e_1^\top P_0'(0) h + e_1^\top \big[P_0(x_0) - P_0(0) - x_0 P_0'(0)\big] h| \\ &\le d_q x_0 + \frac{32 a_q^2 x_0^2}{\alpha_{\text{slow}}^2} \le 5 \varepsilon_{proj}(k). \end{aligned} \tag{109}$$

Since $p_m \in \operatorname{Range} P_0(x_m)$, (73) gives $\|p_m\|_\infty = |e_1^\top p_m|$. Moreover,

$$p_m - p_{m-1} = [P_0(x_m) - P_0(x_{m-1})] p_{m-1}, \qquad \|p_m\|_\infty \le (1 + L_m) \|p_{m-1}\|_\infty. \tag{110}$$

Thus

$$\begin{aligned} \max_{m<k} \|p_m\|_\infty &\overset{(110)}{\le} \prod_{m=1}^{k-1} (1 + L_m) \|p_0\|_\infty \overset{(109)}{\le} e^{\sum_{m=1}^{k-1} L_m} (1 + 5\varepsilon_{proj}(k)) \overset{(108)}{\le} \frac{17}{16}, \\ \|p_{k-1} - p_0\|_\infty &\overset{(110)}{\le} \frac{17}{16} \sum_{m=1}^{k-1} L_m \overset{(108)}{\le} 17 \varepsilon_{proj}(k). \end{aligned} \tag{111}$$

Since $N_q(x) P_0(x) = z_{0,q}(x) P_0(x)$, we have $D_{j,k}^s = -w_j^\star \prod_{m=0}^{k-1} z_{0,q}(x_m) e_1^\top p_{k-1} = e_1^\top p_{k-1} \overline{U}_{j,k}$. Then (109) and (111) imply

$$|D_{j,k}^s - \overline{U}_{j,k}| = |\overline{U}_{j,k}| \, |e_1^\top p_{k-1} - 1| \le |\overline{U}_{j,k}| \left(\|p_{k-1} - p_0\|_\infty + |e_1^\top p_0 - 1|\right) \le 22 \varepsilon_{proj}(k) |\overline{U}_{j,k}|. \tag{112}$$

By (65), $z_{0,q}(x) \le 1 - a_q x \le e^{-a_q x}$. Combining it with (60) gives

$$\left| \sum_{m=0}^{k-1} \log z_{0,q}(x_m) + a_q \lambda T_k \right| \le a_q \left(d_q + \frac{a_q}{2}\right) \lambda^2 \sum_{m=0}^{k-1} \eta_m^2 + c_{\log,q} \lambda^3 \sum_{m=0}^{k-1} \eta_m^3 \le \varepsilon_{exp}(k),$$

where we also use $a_q \le a_{\text{full}}$, $d_q \le d_1$, and $c_{\log,q} \le c_{\log,1}$. It follows that

$$|\overline{U}_{j,k} - U_{j,k}^0| \le \varepsilon_{exp}(k) |U_{j,k}^0| \tag{113}$$

Combining this estimate with (112) and $|\overline{U}_{j,k}| \le |U_{j,k}^0|$ yields

$$|D_{j,k}^s - U_{j,k}^0| \le \big(22 \varepsilon_{proj}(k) + \varepsilon_{exp}(k)\big) |U_{j,k}^0| \le \varepsilon_{ep}(k) |U_{j,k}^0|. \tag{114}$$

Part 2: approximation errors related to $e_1^\top \Phi_{j;k,0} X_{j,0} - D_{j,k}^s$.

Set

$$y_{-1} := h, \qquad y_m := N_q(x_m) y_{m-1}, \quad 0 \le m < k,$$

and for $m \geq 1$ consider the decomposition

$$\begin{aligned} s_m &:= N_q(x_m)P_0(x_m)y_{m-1}, \quad f_m := N_q(x_m)P_1(x_m)y_{m-1}, \\ s_m^0 &:= N_q(x_m)P_0(x_m)s_{m-1}^0, \quad a_m := s_m - s_m^0 \end{aligned}$$

Note that $y_m = s_m + f_m$, and $\{s_m^0\}_m$ generates $D_{j,k}^s$. For $m \geq 1$, we have

$$\begin{aligned} a_m &= N_q(x_m)P_0(x_m)(a_{m-1} + f_{m-1}), \\ f_m &= N_q(x_m)P_1(x_m)(s_{m-1}^0 + a_{m-1} + f_{m-1}). \end{aligned}$$

For any $u \in \operatorname{Range} P_0(x_{m-1})$, $v \in \operatorname{Range} P_1(x_{m-1})$, we have $P_0(x_{m-1})v = 0$ and

$$\begin{aligned} P_0(x_m)u &= u + (P_0(x_m) - P_0(x_{m-1}))u, \quad P_1(x_m)u = (P_0(x_{m-1}) - P_0(x_m))u, \\ P_0(x_m)v &= (P_0(x_m) - P_0(x_{m-1}))v, \qquad P_1(x_m)v = (I - P_0(x_m))v. \end{aligned} \tag{115}$$

Thus

$$\begin{aligned} \|P_0(x_m)u\|_\infty &\leq (1 + L_m)\|u\|_\infty, \quad \|P_1(x_m)u\|_\infty \leq L_m\|u\|_\infty, \\ \|P_0(x_m)v\|_\infty &\leq L_m\|v\|_\infty, \qquad \|P_1(x_m)v\|_\infty \leq (1 + L_m)\|v\|_\infty. \end{aligned} \tag{116}$$

Set

$$Q_m := \frac{\|s_m^0\|_\infty}{\exp(-a_q \sum_{\ell=0}^m x_\ell)}, \qquad A_m := \frac{\|a_m\|_\infty}{\exp(-a_q \sum_{\ell=0}^m x_\ell)}, \qquad F_m := \frac{\|f_m\|_\infty}{\exp(-a_q \sum_{\ell=0}^m x_\ell)}.$$

The inequality $a_q x_m \leq \alpha_{\text{slow}}/8 < 17\alpha_{\text{slow}}/20$ implies $1 - \frac{17}{20}\alpha_{\text{slow}} \leq e^{-a_q x_m}$. Hence combining (116) and (74) yields

$$\begin{aligned} Q_m &\leq (1 + L_m)Q_{m-1}, \\ A_m &\leq (1 + L_m)A_{m-1} + L_m F_{m-1}, \\ F_m &\leq L_m(Q_{m-1} + A_{m-1}) + (1 + L_m)F_{m-1}. \end{aligned} \tag{117}$$

At $m = 0$, the identities $s_0^0 = z_{0,q}(x_0)P_0(x_0)h$, $f_0 = N_q(x_0)P_1(x_0)h$ together with $z_{0,q}(x_0) \leq e^{-a_q x_0}$, (74), and $1 - \frac{17}{20}\alpha_{\text{slow}} \leq e^{-a_q x_0}$ give

$$Q_0 \leq \|P_0(x_0)h\|_\infty, \qquad F_0 \leq \|P_1(x_0)h\|_\infty.$$

Using $\|h\|_\infty = 1$, (71) and (72), we get

$$\begin{aligned} \|P_0(x_0)h\|_\infty &= |e_1^\top P_0(x_0)h| \\ &\leq |x_0 e_1^\top P_0(0)h| + |x_0 e_1^\top P_0'(0)h| + |e_1^\top (P_0(x_0) - P_0(0) - x_0 P_0'(0))h| \\ &\leq 1 + d_q x_0 + \frac{32 a_q^2 x_0^2}{\alpha_{\text{slow}}^2} \leq 1 + 5\varepsilon_{proj}(k) \leq \frac{65}{64}, \end{aligned} \tag{118}$$

where the third inequality also uses $d_q \leq a_q/\alpha_{\text{slow}}$ and $a_q x_0 \leq \frac{\alpha_{\text{slow}}}{8}$. On the other hand, $P_1(0)h = (0, -1, -1)^\top$ implies

$$\|P_1(x_0)h\|_\infty \leq 1 + \|P_0(x_0) - P_0(0)\|_\infty \leq 1 + \frac{16 a_q x_0}{\alpha_{\text{slow}}} \leq \frac{65}{64}.$$

Consequently $\max\{Q_0, F_0\} \leq 65/64$, and hence $Q_m \leq 17/16$. Define $C_m := A_m + F_m$. Then (117) yields

$$C_m \leq (1 + 2L_m)C_{m-1} + L_m Q_{m-1}.$$

Using $S_{0,k} \leq 1/64$ and iterating gives

$$\max_{m<k} C_m \leq e^{2S_{0,k}} \left(\frac{65}{64} + \frac{17}{16} S_{0,k} \right) \leq \frac{7}{6}.$$

For $k \geq 2$, iteration of the second line of (117) gives

$$A_{k-1} \leq e^{S_{0,k}} \max_{m<k} C_m \leq e^{S_{0,k}} \frac{7}{6} S_{0,k} \leq 19\varepsilon_{proj}(k). \tag{119}$$

It also holds for $k=1$ as $A_0=0$. Finally, $e_1^\top P_1(0)=0$, so

$$\|e_1^\top P_1(x_{k-1})\|_1 \le \|P_0(x_{k-1})-P_0(0)\|_\infty \le \frac{16a_q x_{k-1}}{\alpha_{\rm slow}}. \tag{120}$$

Note that

$$e_1^\top \Phi_{j;k,0}X_{j,0} - D^s_{j,k} = -w_j^\star e_1^\top (y_{k-1}-s^0_{k-1}) = -w_j^\star e_1^\top (a_{k-1}+f_{k-1}),$$

and

$$\begin{aligned} &|e_1^\top a_{k-1}| \le \|a_{k-1}\|_\infty, \\ &|e_1^\top f_{k-1}| = |e_1^\top P_1(x_{k-1})f_{k-1}| \le \|e_1^\top P_1(x_{k-1})\|_1 \|f_{k-1}\|_\infty \overset{(120)}{\le} \frac{16a_q x_{k-1}}{\alpha_{\rm slow}}\|f_{k-1}\|_\infty. \end{aligned} \tag{121}$$

Thus

$$\begin{aligned} |e_1^\top \Phi_{j;k,0}X_{j,0} - D^s_{j,k}| &\le |w_j^\star| e^{\sum_{i=0}^{k-1} x_i}\left(A_{k-1} + \frac{16a_q x_{k-1}}{\alpha_{\rm slow}}F_{k-1}\right) \\ &\overset{(119)}{\le} 38\varepsilon_{proj}(k)|U^0_{j,k}| \le \varepsilon_{ep}(k)|U^0_{j,k}|, \end{aligned}$$

where the second inequality also uses $a_q x_{k-1}/\alpha^2_{\rm slow} \le \varepsilon_{proj}(k)$. Combining this estimate with (114) proves

$$|e_1^\top \Phi_{j;k,0}X_{j,0} - U^0_{j,k}| \le 2\varepsilon_{ep}(k)|U^0_{j,k}|. \tag{122}$$

Thus $\left||e_1^\top \Phi_{j;k,0}X_{j,0}|^2 - |U^0_{j,k}|^2\right| \le 4\varepsilon_{ep}(k)\big(1+\varepsilon_{ep}(k)\big)|U^0_{j,k}|^2$. Since $\sum_{j=1}^d \lambda_j |U^0_{j,k}|^2 = S_A(T_k)$, and $\varepsilon_{ep}(k)\le 1/16$, we have

$$\left| e_k^{\rm signal} - \frac{1}{2}S_A(T_k)\right| \le 2\varepsilon_{ep}(k)\big(1+\varepsilon_{ep}(k)\big)S_A(T_k) \le 4\varepsilon_{ep}(k)S_A(T_k).$$

This proves (105).

Part 3: approximation errors related to the noise decay. Here we focus on a fixed coordinate (i.e., fix λ, q), and omit the subscript for notational brevity. For $0\le i\le k-1$, define

$$\begin{aligned} &\mathbf{0}_k := (0,\dots,0)^\top \in \mathbb{R}^k, \qquad \mathbf{e}_i^{(k)} := \big(\underbrace{0,\dots,0}_{i \text{ entries}},1,\ \underbrace{0,\dots,0}_{k-i-1 \text{ entries}}\big)^\top \in \mathbb{R}^k, \\ &\mathbf{u}_0 = \mathbf{0}_k, \qquad \mathbf{m}^{\rm f}_{-1} = \mathbf{m}^{\rm s}_{-1} = \mathbf{0}_k, \qquad \mathbf{g}_i = \lambda \mathbf{u}_i + \mathbf{e}_i^{(k)} \end{aligned} \tag{123}$$

If $[\mathbf{x}]_i$ denotes coordinate i of $\mathbf{x}\in\mathbb{R}^k$, then the preceding definition implies $[\mathbf{g}_t]_i = \lambda[\mathbf{u}_t]_i + \mathbb{I}_{\{i=t\}}$ for $0\le i,t\le k-1$. We further let

$$\begin{aligned} \mathbf{m}^{\rm f}_t &= \rho_{\rm fast}\mathbf{m}^{\rm f}_{t-1} + \mathbf{g}_t, \\ \mathbf{m}^{\rm s}_t &= \rho_{\rm slow}\mathbf{m}^{\rm s}_{t-1} + \mathbf{g}_t, \\ \mathbf{u}_{t+1}-\mathbf{u}_t &= -\eta_t\big(\gamma\mathbf{g}_t + \mathbf{m}^{\rm f}_t + \chi q \mathbf{m}^{\rm s}_t\big), \qquad 0\le t\le k-1. \end{aligned} \tag{124}$$

Then

$$\mathbf{u}_t = \big(R_{j;t,0},\dots,R_{j;t,t-1},\underbrace{0,\dots,0}_{k-t \text{ entries}}\big)^\top = \sum_{i=0}^{t-1} R_{j;t,i}\mathbf{e}_i^{(k)}, \qquad 0\le t\le k, \tag{125}$$

and

$$\mathbf{g}_t = \big(\lambda R_{j;t,0},\dots,\lambda R_{j;t,t-1},1,\ \underbrace{0,\dots,0}_{k-t-1 \text{ entries}}\big)^\top, \qquad 0\le t\le k-1.$$

We also define $\mathbf{g}_r = \mathbf{0}_k$ for $r<0$. Since $\mathbf{m}^{\rm f}_t = \sum_{\ell=0}^t \rho_{\rm fast}^\ell \mathbf{g}_{t-\ell}$, $\mathbf{m}^{\rm s}_t = \sum_{\ell=0}^t \rho_{\rm slow}^\ell \mathbf{g}_{t-\ell}$, we get that for $0\le t\le k-1$,

$$\mathbf{u}_{t+1}-\mathbf{u}_t = -\eta_t\left(\gamma\mathbf{g}_t + \sum_{\ell=0}^t \rho_{\rm fast}^\ell \mathbf{g}_{t-\ell} + \chi q\sum_{\ell=0}^t \rho_{\rm slow}^\ell \mathbf{g}_{t-\ell}\right) = -\eta_t\sum_{\ell=0}^t C_\ell(q)\mathbf{g}_{t-\ell}, \tag{126}$$

where $C_\ell(q) := \gamma \mathbb{I}_{\{\ell=0\}} + \rho_{\text{fast}}^\ell + \chi q \rho_{\text{slow}}^\ell$ for every integer $\ell \geq 0$.

where the series is absolutely convergent because $0 \leq \rho_{\text{fast}}, \rho_{\text{slow}} < 1$. For a $\mathbb{R}_k$-valued sequence $\mathbf{b} = (\mathbf{b}_t)_{t\geq 0}$, define the convolution with $D_n(q) := \sum_{\ell=n}^\infty C_\ell(q) = \frac{\rho_{\text{fast}}^n}{\alpha_{\text{fast}}} + \chi q \frac{\rho_{\text{slow}}^n}{\alpha_{\text{slow}}}$ $(n \geq 1)$ by

$$(D_q * \mathbf{b})_t := \begin{cases} \sum_{n=1}^{t+1} D_n(q)\mathbf{b}_{t-n+1}, & t \geq 0, \\ 0, & t < 0. \end{cases} \tag{127}$$

It follows that for every $t \geq 0$,

$$(D_q * \mathbf{b})_t - (D_q * \mathbf{b})_{t-1} = D_1(q)\mathbf{b}_t + \sum_{\ell=1}^{t} \big[D_{\ell+1}(q) - D_\ell(q)\big]\mathbf{b}_{t-\ell} = a_q \mathbf{b}_t - \sum_{\ell=0}^{t} C_\ell(q)\mathbf{b}_{t-\ell}. \tag{128}$$

Applying (128) with $\mathbf{b} = \mathbf{g}$ turns (126) into

$$\mathbf{u}_{t+1} - \mathbf{u}_t = -a_q \eta_t \mathbf{g}_t + \eta_t \big[(D_q * \mathbf{g})_t - (D_q * \mathbf{g})_{t-1}\big], \qquad 0 \leq t \leq k-1. \tag{129}$$

Thus the effective gain a_q is separated exactly from a discrete boundary difference; no spectral approximation has been made.

Define the sequence $\mathbf{u}_0^E = 0$,

$$\mathbf{u}_{t+1}^E = (1 - a_q \lambda \eta_t)\mathbf{u}_t^E - a_q \eta_t \mathbf{e}_t^{(k)}, \qquad 0 \leq t \leq k-1. \tag{130}$$

It satisfies

$$\mathbf{u}_t^E = -a_q \sum_{i=0}^{t-1} \eta_i \prod_{\ell=i+1}^{t-1} (1 - a_q \lambda \eta_\ell)\mathbf{e}_i^{(k)}, \qquad 0 \leq t \leq k.$$

Define

$$\begin{aligned} \mathbf{g}_t^E &:= \lambda \mathbf{u}_t^E + \mathbf{e}_t^{(k)}, \quad 0 \leq t \leq k-1, \\ \mathbf{v}_t &:= \mathbf{u}_t - \mathbf{u}_t^E, \qquad 0 \leq t \leq k, \end{aligned}$$

and set $\mathbf{g}_k^E := 0$. Let $\mathsf{X}_k := (\mathbb{R}_k)^{\{0,\dots,k\}}$. Define the map $\mathsf{B}_q : \mathsf{X}_k \to \mathsf{X}_k$ as $(\mathsf{B}_q \mathbf{b})_0 := 0$,

$$(\mathsf{B}_q \mathbf{b})_{t+1} := (1 - a_q \lambda \eta_t)(\mathsf{B}_q \mathbf{b})_t + \eta_t \big[(D_q * \mathbf{b})_t - (D_q * \mathbf{b})_{t-1}\big], \qquad 0 \leq t \leq k-1. \tag{131}$$

Since $\mathbf{g}_t - \mathbf{g}_t^E = \lambda \mathbf{v}_t$ for $0 \leq t \leq k-1$, subtracting (130) from (129) gives that for $0 \leq t \leq k-1$,

$$\mathbf{v}_{t+1} = (1 - a_q \lambda \eta_t)\mathbf{v}_t + \eta_t \big[(D_q * \mathbf{g}^E)_t - (D_q * \mathbf{g}^E)_{t-1}\big] + \lambda \eta_t \big[(D_q * \mathbf{v})_t - (D_q * \mathbf{v})_{t-1}\big].$$

Thus $\mathbf{v} \in \mathsf{X}_k$ satisfies

$$\mathbf{v} = \mathsf{B}_q \mathbf{g}^E + \lambda \mathsf{B}_q \mathbf{v}. \tag{132}$$

Thus we consider the decomposition $\mathbf{v}_t = \sum_{r=1}^t \mathbf{v}_t^{(r)}$, where $\mathbf{v}^{(r+1)} := \lambda \mathsf{B}_q \mathbf{v}^{(r)}$ $(1 \leq r < k)$ and $\mathbf{v}^{(1)} := \mathsf{B}_q \mathbf{g}^E$. We set $\mathbf{v}_0 = 0$.

We first bound B_q for a general input. Set $\gamma_{m,t} := \eta_t \prod_{\ell=t+1}^{m-1}(1 - a_q \lambda \eta_\ell)$ for $0 \leq t < m \leq k$. Then (131) gives

$$(\mathsf{B}_q \mathbf{b})_m = \sum_{t=0}^{m-1} \gamma_{m,t} \big[(D_q * \mathbf{b})_t - (D_q * \mathbf{b})_{t-1}\big].$$

Since $(D_q * \mathbf{b})_{-1} = 0$, summation by parts yields

$$(\mathsf{B}_q \mathbf{b})_m = \eta_{m-1}(D_q * \mathbf{b})_{m-1} + \sum_{t=0}^{m-2} (\gamma_{m,t} - \gamma_{m,t+1})(D_q * \mathbf{b})_t. \tag{133}$$

Then by $\sum_{n=1}^\infty D_n(q) = d_q$, we get

$$\|(\mathsf{B}_q \mathbf{b})_m\| \leq (\eta_{m-1} + \sum_{t=0}^{m-2} |\gamma_{m,t} - \gamma_{m,t+1}|) d_q \sup_{0 \leq t \leq m-1} \|\mathbf{b}_t\|. \tag{134}$$

Substituting

$$\begin{aligned}\sum_{t=0}^{m-2}|\gamma_{m,t+1}-\gamma_{m,t}| =& \sum_{t=0}^{m-2}\left|\left(\prod_{\ell=t+2}^{m-1}(1-a_q\lambda\eta_\ell)\right)(\eta_{t+1}-\eta_t)+a_q\lambda\eta_t\eta_{t+1}\prod_{\ell=t+2}^{m-1}(1-a_q\lambda\eta_\ell)\right| \\ \le& V_k(\eta)+\eta_{\max,k}\sum_{t=0}^{m-2}\Big(\prod_{\ell=t+2}^{m-1}(1-a_q\lambda\eta_\ell)-\prod_{\ell=t+1}^{m-1}(1-a_q\lambda\eta_\ell)\Big) \\ \le& V_k(\eta)+\eta_{\max,k}.\end{aligned} \tag{135}$$

into (134) yields

$$\sup_{0\le m\le k}\|(\mathsf{B}_q\mathbf{b})_m\| \le 2\big(\eta_{\max,k}+V_k(\eta)\big)d_q\sup_{0\le t\le k-1}\|\mathbf{b}_t\|. \tag{136}$$

Set $y_i := a_q\lambda\eta_i$ and note that

$$G_q^2=\sum_{n=1}^{\infty}D_n(q)^2=\frac{\rho_{\text{fast}}^2}{\alpha_{\text{fast}}^2(1-\rho_{\text{fast}}^2)}+\frac{2\chi q\rho_{\text{fast}}\rho_{\text{slow}}}{\alpha_{\text{fast}}\alpha_{\text{slow}}(1-\rho_{\text{fast}}\rho_{\text{slow}})}+\frac{\chi^2q^2\rho_{\text{slow}}^2}{\alpha_{\text{slow}}^2(1-\rho_{\text{slow}}^2)}\le G_1^2.$$

Direct computation gives

$$\sum_{n=1}^{\infty}\frac{\rho_{\text{fast}}^n}{\alpha_{\text{fast}}}\le\sqrt{\frac{2}{\alpha_{\text{slow}}}}\left(\sum_{n=1}^{\infty}\frac{\rho_{\text{fast}}^{2n}}{\alpha_{\text{fast}}^2}\right)^{1/2},\qquad\sum_{n=1}^{\infty}\frac{\chi q\rho_{\text{slow}}^n}{\alpha_{\text{slow}}}\le\sqrt{\frac{2}{\alpha_{\text{slow}}}}\left(\sum_{n=1}^{\infty}\frac{\chi^2q^2\rho_{\text{slow}}^{2n}}{\alpha_{\text{slow}}^2}\right)^{1/2},$$

implying that $d_q=\sum_{n=1}^{\infty}D_n(q)\le\frac{2G_q}{\sqrt{\alpha_{\text{slow}}}}$. By (130), we have that for $1\le t\le k$,

$$\lambda\mathbf{u}_t^E=-\sum_{i=0}^{t-1}y_i\prod_{\ell=i+1}^{t-1}(1-y_\ell)\mathbf{e}_i^{(k)}.$$

Then $0\le y_i\le\alpha_{\text{slow}}/8$ gives that for $1\le t\le k$

$$\begin{aligned}\|\lambda\mathbf{u}_t^E\| =& \sum_{i=0}^{t-1}y_i^2\prod_{\ell=i+1}^{t-1}(1-y_\ell)^2\le\left(\max_{0\le i\le t-1}y_i\right)\sum_{i=0}^{t-1}\big[1-(1-y_i)^2\big]\prod_{\ell=i+1}^{t-1}(1-y_\ell)^2 \\ =& \left(\max_{0\le i\le t-1}y_i\right)\left[1-\prod_{\ell=0}^{t-1}(1-y_\ell)^2\right]\le\frac{\alpha_{\text{slow}}}{8}.\end{aligned} \tag{137}$$

For $0\le t\le k-1$,

$$(D_q*\mathbf{g}^E)_t=\sum_{n=1}^{t+1}D_n(q)\mathbf{e}_{t-n+1}^{(k)}+\sum_{n=1}^{t+1}D_n(q)\lambda\mathbf{u}_{t-n+1}^E.$$

Hence for any $0\le t\le k-1$,

$$\begin{aligned}\|(D_q*\mathbf{g}^E)_t\| \le& \left\|\sum_{n=1}^{t+1}D_n(q)\mathbf{e}_{t-n+1}^{(k)}\right\|+\sum_{n=1}^{t+1}D_n(q)\|\lambda\mathbf{u}_{t-n+1}^E\| \\ =& \left(\sum_{n=1}^{t+1}D_n(q)^2\right)^{1/2}+\sum_{n=1}^{t+1}D_n(q)\|\lambda\mathbf{u}_{t-n+1}^E\| \\ \le& G_q+d_q\sup_{0\le s\le t}\|\lambda\mathbf{u}_s^E\| \\ \overset{(137)}{\le}& \left(1+\frac{1}{\sqrt{2}}\right)G_q.\end{aligned} \tag{138}$$

Since $(\mathbf{v}^{(1)})_m=(\mathsf{B}_q\mathbf{g}^E)_m=\eta_{m-1}(D_q*\mathbf{g}^E)_{m-1}+\sum_{t=0}^{m-2}(\gamma_{m,t}-\gamma_{m,t+1})(D_q*\mathbf{g}^E)_t$, combining (133), (135), and (138) yields

$$\|\mathbf{v}_m^{(1)}\|\le\left(\eta_{m-1}+\sum_{t=0}^{m-2}|\gamma_{m,t}-\gamma_{m,t+1}|\right)\sup_{0\le t<m}\|(D_q*\mathbf{g}^E)_t\|\le(2+\sqrt{2})\big(\eta_{\max,k}+V_k(\eta)\big)G_q. \tag{139}$$

Set $\theta_k := 2\lambda\big(\eta_{\max,k} + V_k(\eta)\big)d_q$. Then $\theta_k \le 2\alpha_{\text{slow}}\varepsilon_{proj}(k) \le 2\varepsilon_{proj}(k) \le \frac{1}{512}$ by $d_q \le a_q/\alpha_{\text{slow}}$ and $a_q\lambda \le a_{\text{full}}\lambda_1$. (136) gives that for $1 \le r < k$,

$$\sup_{0\le m\le k}\|\mathbf{v}_m^{(r+1)}\| \le \theta_k \sup_{0\le m\le k}\|\mathbf{v}_m^{(r)}\|.$$

Combining this with (139) yields

$$\begin{aligned}\sup_{0\le m\le k}\|\mathbf{v}_m\| &\le \sum_{r=1}^{k}\sup_{0\le m\le k}\|\mathbf{v}_m^{(r)}\| \le (2+\sqrt{2})\big(\eta_{\max,k}+V_k(\eta)\big)G_q\sum_{r=1}^{k}\theta_k^{r-1}\\ &\le \frac{2+\sqrt{2}}{1-\theta_k}\big(\eta_{\max,k}+V_k(\eta)\big)G_q.\end{aligned}$$

Substituting it and (134) into (132) yields a finer estimate:

$$\begin{aligned}\|\mathbf{v}_k\| &\le \left((1+\frac{1}{\sqrt{2}})G_q + \lambda d_q \sup_{0\le m\le k}\|\mathbf{v}_m\|\right)(\eta_{k-1}+\sum_{t=0}^{k-2}|\gamma_{k,t}-\gamma_{k,t+1}|)\\ &\le 2G_q(\eta_{k-1}+\sum_{t=0}^{k-2}|\gamma_{k,t}-\gamma_{k,t+1}|).\end{aligned}$$

Define $R^0_{j;k,i} := -\eta_i a_q e^{-a_q\lambda(T_k-T_i)}$ and $R^E_{j;k,i} := -a_q\eta_i\prod_{\ell=i+1}^{k-1}(1-a_q\lambda\eta_\ell)$ for $0\le i\le k-1$. Note that (125) gives

$$\mathbf{v}_k = \sum_{i=0}^{k-1}\big(R_{j;k,i}-R^E_{j;k,i}\big)\mathbf{e}_i^{(k)}.$$

Then

$$\sum_{i=0}^{k-1}(R_{j;k,i}-R^E_{j;k,i})^2 = \|\mathbf{v}_k\|^2 \le 4(\eta_{k-1}+\sum_{t=0}^{k-2}|\gamma_{k,t}-\gamma_{k,t+1}|)^2G_1^2. \tag{140}$$

Next we bound $|R^E_{j;k,i}-R^0_{j;k,i}|$. Note that

$$y_i \le a_q\lambda\eta_{\max,k} \le \alpha_{\text{slow}}^2\varepsilon_{proj}(k) \le \varepsilon_{proj}(k),$$

and $\sum_{\ell=i+1}^{k-1}y_\ell^2 \le 2\varepsilon_{exp}(k)$ as $a_q^2\lambda^2 \le a_{\text{full}}^2\lambda_1^2 \le 2a_{\text{full}}(d_1+a_{\text{full}}/2)\lambda_1^2$. Combining them with $0 \le -\log(1-y)-y \le \frac{4}{7}y^2$ for $0\le y\le 1/8$, we get

$$\log\frac{|R^E_{j;k,i}|}{|R^0_{j;k,i}|} = y_i + \sum_{\ell=i+1}^{k-1}\big[\log(1-y_\ell)+y_\ell\big] \le \varepsilon_{proj}(k)+\frac{8}{7}\varepsilon_{exp}(k) \le \frac{1}{7}\varepsilon_{ep}(k).$$

Using $|e^x-1|\le 2|x|$ for $|x|\le 1/2$, we obtain

$$|R^E_{j;k,i}-R^0_{j;k,i}| \le \varepsilon_{ep}(k)|R^0_{j;k,i}|. \tag{141}$$

Taking summation of (140), (141) over the coordinate j gives

$$\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}-R^E_{j;k,i})^2 s_i \le \widehat{\delta}(k)\max_{0\le i<k}s_i = (\widehat{\mathcal{B}}s)_k, \tag{142}$$

and

$$\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R^E_{j;k,i}-R^0_{j;k,i})^2 s_i \le \varepsilon_{ep}(k)^2(\mathcal{T}s)_k, \tag{143}$$

respectively. Note that

$$\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R^0_{j;k,i})^2 s_i = (\mathcal{T}s)_k, \qquad \sum_{i=0}^{k-1}K^{\text{noise}}_{k,i}s_i = \sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i})^2 s_i. \tag{144}$$

Thus

$$\begin{aligned}\left|(\mathcal{T}s)_k-\sum_{i=0}^{k-1}K^{\text{noise}}_{k,i}s_i\right|^2 &\le \sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}+R^0_{j;k,i})|R_{j;k,i}-R^0_{j;k,i}|s_i\\ &\le\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}+R^0_{j;k,i})^2 s_i}\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}-R^0_{j;k,i})^2 s_i}\\ &\le\big((2+\varepsilon_{ep}(k))\sqrt{(\mathcal{T}s)_k}+\sqrt{(\widehat{\mathcal{B}}s)_k}\big)\big(\varepsilon_{ep}(k)\sqrt{(\mathcal{T}s)_k}+\sqrt{(\widehat{\mathcal{B}}s)_k}\big)\\ &=\big(2\varepsilon_{ep}(k)+\varepsilon_{ep}(k)^2\big)(\mathcal{T}s)_k+2(1+\varepsilon_{ep}(k))\sqrt{(\mathcal{T}s)_k(\widehat{\mathcal{B}}s)_k}+(\widehat{\mathcal{B}}s)_k,\end{aligned} \tag{145}$$

where the second inequality uses Cauchy–Schwarz Inequality, the third inequality uses

$$\begin{aligned}\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}-R^0_{j;k,i})^2 s_i} &\le\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}-R^{\mathrm{E}}_{j;k,i})^2 s_i}+\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R^{\mathrm{E}}_{j;k,i}-R^0_{j;k,i})^2 s_i}\\ &\overset{(142),(143)}{\le}\sqrt{(\widehat{\mathcal{B}}s)_k}+\varepsilon_{ep}(k)\sqrt{(\mathcal{T}s)_k},\end{aligned} \tag{146}$$

and

$$\begin{aligned}\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}+R^0_{j;k,i})^2 s_i} &\le\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R_{j;k,i}-R^0_{j;k,i})^2 s_i}+2\sqrt{\sum_{i=0}^{k-1}\sum_{j=1}^{d}\lambda_j^2(R^0_{j;k,i})^2 s_i}\\ &\overset{(146),(144)}{\le}\sqrt{(\widehat{\mathcal{B}}s)_k}+(2+\varepsilon_{ep}(k))\sqrt{(\mathcal{T}s)_k}.\end{aligned} \tag{147}$$

This proves (106) as $\varepsilon_{ep}(k)\le 1/16$.

□

We define the following notations for proving Theorem 7.2:

$$\begin{aligned}&\mathsf{F}(k):=\mathsf{S}(k)+\sigma^2\sum_{i=0}^{k-1}\eta_i^2\mathsf{K}(T_k-T_i),\\ &c_{sk,-}:=\min\{1,c_{S,-},c_{K,-}\},\qquad c_{sk,+}:=\max\{1,c_{S,+},c_{K,+}\},\\ &\widehat{\delta}^{\max}(k):=\max_{1\le n<k}\widehat{\delta}(n),\qquad K_\eta^{\max}(k):=\max_{1\le n<k}\sum_{i=0}^{n-1}\eta_i^2K_A(T_n-T_i),\qquad \widehat{\varepsilon}(k):=\frac{\sigma^2\widehat{\delta}(k)}{\mathsf{F}(k)},\end{aligned} \tag{148}$$

where we assume $\mathbb{E}\mathcal{E}(w_0)\le C_0\sigma^2$ with C_0 treated as a constant. At $k=1$, both maxima over $1\le n<k$ in (148) are defined to be zero, and $K_\eta^{\max}(1):=0$.

Assumption I.10 (Admissible hyper-parameters)**.** Suppose that the horizon k and the learning rate schedule $\{\eta_i\}$ satisfy

$$\begin{aligned}&\varepsilon_{ep}(k)\le\frac{1}{16},\qquad K_\eta^{\max}(k)\le\frac{1}{8},\qquad \widehat{\delta}^{\max}(k)\le\frac{1}{25},\\ &\widehat{\varepsilon}(k)\le\frac{c_{sk,-}}{16(6C_0+1)},\qquad a_{\text{full}}\lambda_1\eta_{\max,k}\le\frac{\alpha_{\text{slow}}}{8}.\end{aligned} \tag{149}$$

Remark I.11 (Simplified sufficient conditions for admissible hyperparameters)**.** Here we provide a cleaner version of the hyperparameter constraints in (149). The constraints on $\varepsilon_{ep}(k)$ and $\widehat{\delta}^{\max}(k)$ reduce to upper bounds on $\eta_{\max,k}$, $\sum_{i=1}^{k-2}|\eta_{i+1}-\eta_i|$, and $\max_{i<k-1}|\eta_{i+1}/\eta_i-1|$. By (153), a sufficient condition for $K_\eta^{\max}(k)\le\frac{1}{8}$ is $\eta_{\max,k}\le\frac{1}{4}\left(a_0\sum_{j=1}^{r}j^{-\nu}+a_1\sum_{j=r+1}^{d}j^{-\nu}\right)^{-1}$, which also becomes an upper bound constraint to $\eta_{\max,k}$.

We next turn to the upper bound constraint on $\widehat{\varepsilon}(k)$. Deriving a simplified sufficient condition for general learning rate schedules is challenging. However, for some common choices such as constant and power-law decay, it can be satisfied. Excluding the very early stage, we consider the regime $T_k \gtrsim \frac{r^\nu}{a_{\text{top}}}$ so that $\mathsf{K}(T_k - s) \lesssim \mathsf{K}(T_k - T_i)$ for $s \in [T_i, T_{i+1}]$. Hence,

$$\int_0^{T_k} \mathsf{K}(z)\, dz = \sum_{i=0}^{k-1} \int_{T_i}^{T_{i+1}} \mathsf{K}(T_k - s)\, ds \lesssim \sum_{i=0}^{k-1} \eta_i \mathsf{K}(T_k - T_i) \le \frac{1}{\eta_{k-1}} \sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i). \quad (150)$$

This yields $\sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i) \gtrsim \eta_{k-1}$. Now consider $\eta_i = \eta_0(1 + i/\tau_0)^{-\gamma}$, where $\tau_0 \in \mathbb{R}_+ \cup \{+\infty\}$ and $1/2 < \gamma \le 1$. The inequality $\frac{\eta_t - \eta_{t+1}}{\eta_{t+1}} = \left(1 + \frac{1}{\tau_0 + t}\right)^\gamma - 1 \le \frac{\gamma}{\tau_0 + t}$ gives

$$\left[\sum_{t=0}^{k-2} (\eta_t - \eta_{t+1}) \sqrt{\mathsf{K}(T_k - T_{t+2})}\right]^2 \lesssim \sum_{t=0}^{k-2} \left(\frac{\eta_t - \eta_{t+1}}{\eta_{t+1}}\right)^2 \sum_{t=0}^{k-2} \eta_{t+1}^2 \mathsf{K}(T_k - T_{t+1}) \le \frac{2\gamma^2}{\tau_0} \sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i). \quad (151)$$

Combining this with

$$\frac{\eta_{k-1}^2 \mathsf{K}(0)}{\sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i)} \lesssim \eta_{k-1},$$

we obtain

$$\frac{\left[\eta_{k-1}\sqrt{\mathsf{K}(0)} + \sum_{t=0}^{k-2} (\eta_t - \eta_{t+1}) \sqrt{\mathsf{K}(T_k - T_{t+2})}\right]^2}{\sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i)} \lesssim \eta_{k-1} + \frac{\gamma^2}{\tau_0}.$$

Note that $V_k(\eta) \le \eta_0$, $\delta_{\text{step}}(k) \le \frac{\gamma}{\alpha_{\text{slow}} \tau_0}$. By Lemma I.13, for sufficiently large d, τ_0 and sufficiently small η_0, the admissibility condition (149) holds.

Remark I.12 (Scope of T_k)**.** The regime $T_k \lesssim d^\nu / a_{\text{full}}$ already encompasses the practically typical range $k \lesssim d$. For $T_k \gg d^\nu / a_{\text{full}}$, matching upper and lower bounds for $\mathbb{E}\mathcal{E}(w_k)$ are not available within our current analysis. The same difficulty also arises in the SDE analysis of analogous scenarios of (Li et al., 2025a).

Proof of Theorem 7.2. Suppose the conditions in (149) hold. We structure the proof as follows.

Step 1: the uniform upper bound for $\mathcal{E}$.

For $0 \le i \le k-1$, we define the sequence $\mathcal{E}_i := \mathbb{E}\mathcal{E}(w_i)$, $s_i^- := \mathcal{E}_i + \frac{\sigma^2}{2}$, $s_i^+ := 2\mathcal{E}_i + \frac{\sigma^2}{2}$, and we denote by $\mathbf{1}$ the sequence consisting entirely of ones.

Since K_A is decreasing,

$$\begin{aligned}(\mathcal{T}\mathbf{1})_n &= \sum_{i=0}^{n-1} \eta_i^2 K_A(T_n - T_i) \le \eta_{\max,k} \sum_{i=0}^{n-1} \int_{T_i}^{T_{i+1}} K_A(T_n - s)\, \mathrm{d}s \\ &= \eta_{\max,k} \int_0^{T_n} K_A(u)\, \mathrm{d}u \overset{(81)}{\le} \frac{\eta_{\max,k}}{2} \operatorname{tr}(AH).\end{aligned} \quad (152)$$

Taking the maximum over $0 \le n < k$ gives

$$\max_{0 \le n < k} (\mathcal{T}\mathbf{1})_n \le K_\eta^{\max}(k) \le \frac{\eta_{\max,k}}{2} \operatorname{tr}(AH). \quad (153)$$

The upper bounds in (97) and (107) give that for $1 \le n \le k$,

$$\begin{aligned}\mathcal{E}_n \le& e_n^{\text{signal}} + \sum_{i=0}^{n-1} K_{n,i}^{\text{noise}} s_i^+ \le e_n^{\text{signal}} + \frac{289}{128} (\mathcal{T} s^+)_n + 2(\widehat{\mathcal{B}} s^+)_n \\ \le& \frac{3}{2} C_0 \sigma^2 + \frac{289}{64} (\mathcal{T}\mathcal{E})_n + \frac{289}{256} \sigma^2 (\mathcal{T}\mathbf{1})_n + 4\widehat{\delta}(n) \max_{0 \le i \le n-1} \mathcal{E}_i + \widehat{\delta}(n)\sigma^2.\end{aligned} \quad (154)$$

where the last inequality uses $(\mathcal{T}s^+)_n = 2(\mathcal{T}\mathcal{E})_n + \frac{\sigma^2}{2}(\mathcal{T}\mathbf{1})_n$, $(\widehat{\mathcal{B}}s^+)_n = \widehat{\delta}(n)\left(2\max_{0\le i\le n-1}\mathcal{E}_i + \frac{\sigma^2}{2}\right)$ and

$$e_n^{\text{signal}} \overset{(105)}{\le} \left(\frac{1}{2} + 4\varepsilon_{ep}(n)\right) S_A(T_n) \le \frac{3}{4}S_A(0) \le \frac{3}{2}C_0\sigma^2. \tag{155}$$

Then using $\max_{0\le n<k}(\mathcal{T}\mathcal{E})_n \le K_\eta^{\max}(k)\max_{0\le i\le k-1}\mathcal{E}_i$ and (153) to (154), we get

$$\max_{0\le i\le k-1}\mathcal{E}_i \le \frac{3}{2}C_0\sigma^2 + \left[\frac{289}{64}K_\eta^{\max}(k) + 4\widehat{\delta}^{\max}(k)\right]\left(\max_{0\le i\le k-1}\mathcal{E}_i + \frac{\sigma^2}{4}\right). \tag{156}$$

With the condition $\frac{289}{64}K_\eta^{\max}(k) + 4\widehat{\delta}^{\max}(k) \le \frac{3}{4}$ in (149), we get

$$\max_{0\le i<k}\mathcal{E}_i \le \left(6C_0 + \frac{3}{4}\right)\sigma^2. \tag{157}$$

Step 2: the asymptotic bounds for $\mathcal{E}_k$.

The estimate in (157) gives $\max_{0\le i<k} s_i^- \le \left(6C_0 + \frac{5}{4}\right)\sigma^2$, $\max_{0\le i<k} s_i^+ \le 2\left(6C_0+1\right)\sigma^2$.

Thus

$$\begin{aligned}
(\mathcal{T}s^-)_k &= \sum_{i=0}^{k-1}\eta_i^2 K_A(T_k - T_i)\left(\mathcal{E}_i + \frac{\sigma^2}{2}\right) \ge \frac{\sigma^2}{2}(\mathcal{T}\mathbf{1})_k,\\
(\mathcal{T}s^+)_k &= \sum_{i=0}^{k-1}\eta_i^2 K_A(T_k - T_i)\left(2\mathcal{E}_i + \frac{\sigma^2}{2}\right) \le 2\left(6C_0+1\right)\sigma^2(\mathcal{T}\mathbf{1})_k,\\
(\widehat{\mathcal{B}}s^-)_k &= \widehat{\delta}(k)\max_{0\le i<k} s_i^- \le \left(6C_0 + \frac{5}{4}\right)\sigma^2\widehat{\delta}(k),\\
(\widehat{\mathcal{B}}s^+)_k &= \widehat{\delta}(k)\max_{0\le i<k} s_i^+ \le 2\left(6C_0+1\right)\sigma^2\widehat{\delta}(k)
\end{aligned} \tag{158}$$

Besides, (105) gives

$$\frac{1}{4}S_A(T_k) \le \left(\frac{1}{2} - 4\varepsilon_{ep}(k)\right)S_A(T_k) \le e_k^{\text{signal}} \le \left(\frac{1}{2} + 4\varepsilon_{ep}(k)\right)S_A(T_k) \le \frac{3}{4}S_A(T_k). \tag{159}$$

Define $G_k := S_A(T_k) + \sigma^2(\mathcal{T}\mathbf{1})_k = S_A(T_k) + \sigma^2\sum_{i=0}^{k-1}\eta_i^2 K_A(T_k - T_i)$. The estimates in (80) give

$$\begin{aligned}
G_k &\ge \min\{1, c_{S,-}\}\mathsf{S}(T_k) + \sigma^2\min\{1, c_{K,-}\}\sum_{i=0}^{k-1}\eta_i^2\mathsf{K}(T_k - T_i) \ge c_{sk,-}\mathsf{F}(k),\\
G_k &\le \max\{1, c_{S,+}\}\mathsf{S}(T_k) + \sigma^2\max\{1, c_{K,+}\}\sum_{i=0}^{k-1}\eta_i^2\mathsf{K}(T_k - T_i) \le c_{sk,+}\mathsf{F}(k).
\end{aligned} \tag{160}$$

we obtain

$$\begin{aligned}
\mathcal{E}_k &\overset{(97)}{\ge} e_k^{\text{signal}} + \sum_{i=0}^{k-1}K_{k,i}^{\text{noise}}s_i^- \overset{(107)}{\ge} \frac{1}{4}S_A(T_k) + \frac{225}{512}(\mathcal{T}s^-)_k - (\widehat{\mathcal{B}}s^-)_k\\
&\overset{(158)}{\ge} \frac{1}{4}S_A(T_k) + \frac{225}{1024}\sigma^2(\mathcal{T}\mathbf{1})_k - \left(6C_0 + \frac{5}{4}\right)\sigma^2\widehat{\delta}(k)\\
&\overset{(149)}{\ge} \left[\frac{225}{1024} - \frac{24C_0+5}{64(6C_0+1)}\right]c_{sk,-}G_k = \frac{966C_0+145}{1024(6C_0+1)}c_{sk,-}G_k.
\end{aligned} \tag{161}$$

and

$$\begin{aligned}
\mathcal{E}_k &\overset{(97)}{\le} e_k^{\text{signal}} + \sum_{i=0}^{k-1}K_{k,i}^{\text{noise}}s_i^+ \overset{(107)}{\le} \frac{3}{4}S_A(T_k) + \frac{289}{128}(\mathcal{T}s^+)_k + 2(\widehat{\mathcal{B}}s^+)_k\\
&\overset{(158)}{\le} \frac{3}{4}S_A(T_k) + \frac{289}{64}(6C_0+1)\sigma^2(\mathcal{T}\mathbf{1})_k + 4(6C_0+1)\sigma^2\widehat{\delta}(k) \overset{(149)}{\le} \left(\frac{289}{64}(6C_0+1)c_{sk,+} + \frac{c_{sk,-}}{4}\right)G_k.
\end{aligned} \tag{162}$$

Combining (161), (162) and (160), we get

$$\mathcal{E}_k \asymp \mathsf{F}(k).$$

This proves the conclusion. □

Lemma I.13 (Admissibility for a decreasing schedule)**.** *Suppose Assumption I.1 and $d \geq 2(r+1)$ hold. Let $k \geq 2$, and suppose*

$$\eta_0 \geq \eta_1 \geq \cdots \geq \eta_{k-1} > 0, \qquad a_{\mathrm{full}} T_k \leq C_d d^{\nu}, \qquad a_{\mathrm{full}} \lambda_1 \eta_0 \leq \frac{\alpha_{\mathrm{slow}}}{8}.$$

Then

$$\widehat{\delta}^{\max}(k) \leq 16\eta_0^2 \sum_{j=1}^{d} \lambda_j^2 G_{q_j}^2, \qquad K_\eta^{\max}(k) \leq \frac{\eta_0}{2} \operatorname{tr}(AH). \tag{163}$$

If we further assume

$$\begin{aligned} &\max\left\{\frac{G_0^2}{a_{\mathrm{top}}^2}, \frac{G_1^2}{a_{\mathrm{full}}^2}\right\} \left[\eta_{k-1}\sqrt{\mathsf{K}(0)} + \sum_{t=0}^{k-2}(\eta_t - \eta_{t+1})\sqrt{\mathsf{K}(T_k - T_{t+2})}\right]^2 \\ &\leq \frac{c_{sk,-}}{256(6C_0+1)\max\{1, c_{K,+}\}} \sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i), \end{aligned} \tag{164}$$

it holds that

$$\widehat{\varepsilon}(k) \leq \frac{c_{sk,-}}{16(6C_0+1)}.$$

Proof. For every $1 \leq j \leq d$, $0 \leq t \leq k-1$, and $1 \leq \ell \leq k$, we have

$$0 \leq a_{q_j}\lambda_j \eta_t \leq a_{\mathrm{full}}\lambda_1\eta_0 \leq \frac{\alpha_{\mathrm{slow}}}{8} < 1, \qquad 0 \leq \Pi_{j;k,\ell} \leq 1. \tag{165}$$

Using $\Pi_{j;k,t+2} - \Pi_{j;k,t+1} = a_{q_j}\lambda_j \eta_{t+1} \Pi_{j;k,t+2}$, we get

$$\sum_{t=0}^{k-2} a_{q_j}\lambda_j \eta_t \eta_{t+1} \Pi_{j;k,t+2} = \eta_{k-1} + \sum_{t=0}^{k-2} \Pi_{j;k,t+2}(\eta_t - \eta_{t+1}) - \eta_0 \Pi_{j;k,1}. \tag{166}$$

Combining this with $\Pi_{j;k,t+2} \leq e^{-a_{q_j}\lambda_j (T_k - T_{t+2})}$ yields

$$\begin{aligned} L_{j,k} &\leq 2\left[\eta_{k-1} + \sum_{t=0}^{k-2} \Pi_{j;k,t+2}(\eta_t - \eta_{t+1}) - \frac{\eta_0}{2}\Pi_{j;k,1}\right] \\ &\leq 2\left[\eta_{k-1} + \sum_{t=0}^{k-2}(\eta_t - \eta_{t+1}) e^{-a_{q_j}\lambda_j(T_k - T_{t+2})}\right]. \end{aligned} \tag{167}$$

For $0 \leq u \leq T_k$, (80) implies

$$\sum_{j=1}^{d} \lambda_j^2 G_{q_j}^2 e^{-2a_{q_j}\lambda_j u} \leq \max\left\{\frac{G_0^2}{a_{\mathrm{top}}^2}, \frac{G_1^2}{a_{\mathrm{full}}^2}\right\} \max\{1, c_{K,+}\} \mathsf{K}(u).$$

Combing it with (167) gives

$$\widehat{\delta}(k) \leq 16 \max\left\{\frac{G_0^2}{a_{\mathrm{top}}^2}, \frac{G_1^2}{a_{\mathrm{full}}^2}\right\} \max\{1, c_{K,+}\} \left[\eta_{k-1}\sqrt{\mathsf{K}(0)} + \sum_{t=0}^{k-2}(\eta_t - \eta_{t+1})\sqrt{\mathsf{K}(T_k - T_{t+2})}\right]^2. \tag{168}$$

Note that $L_{j,n} \leq 2\eta_0$ by (167) for $n \leq k$. Therefore $\widehat{\delta}^{\max}(k) \leq 16\eta_0^2 \sum_{j=1}^{d} \lambda_j^2 G_{q_j}^2$, and then we prove (163) by using (153). Finally, using $\mathsf{F}(k) \geq \sigma^2 \sum_{i=0}^{k-1} \eta_i^2 \mathsf{K}(T_k - T_i)$, (168) and (164) proves the conclusion. □

I.2 Acceleration of Flat-Direction Multiscale Momentum under Constant Learning Rates

Corollary I.14 (Constant Learning Rate)**.** *Let $\eta_i \equiv \eta$ and $r \geq 4$. Suppose the conditions of Theorem 7.2 hold. If we further have*

$$a_{\text{top}}^{-1} r^{\nu}(1 + \log(a_{\text{full}}/a_{\text{top}})) \lesssim \eta k \lesssim a_{\text{full}}^{-1} d^{\nu}, \tag{169}$$

then

$$\mathbb{E}\mathcal{E}(w_k) \asymp (a_{\text{full}}\eta k)^{-s} + \eta\sigma^2 \left(a_{\text{top}} \sum_{j=1}^{r} j^{-\nu} + a_{\text{full}} \sum_{j=r+1}^{d} j^{-\nu} \right) \triangleq \mathcal{E}_{\chi}(\eta; k). \tag{170}$$

Remark I.15. According to the constraints in Corollary I.14 and Theorem 7.2, the optimal learning rate $\eta_{\chi,k}^{\star} = \operatorname{argmin}_{\eta} \mathcal{E}_{\chi}(\eta; k)$ is admissible if k satisfies $\frac{r^{\nu}}{a_0}(1 + \log \kappa_{\chi}) \lesssim \left(\frac{k}{\sigma^2 N_{\chi}(a_{\chi})^{s}} \right)^{\frac{1}{s+1}} \lesssim \frac{d^{\nu}}{a_{\chi}}$ and d is sufficiently large.

Proof of Corollary I.14. Set $t_k := k\eta,\ R := r + 1$.

Part 1: estimate the noise accumulation term. Enlarge the constant C_h supplied by Lemma I.7 (iv) so that

$$2\ell_{-}C_h \geq \log 2, \qquad C_h \left(\frac{4}{5} \right)^{\nu} \geq \max\left\{ 1, 2^{1/(1-\frac{1}{\nu})} - 1 \right\}. \tag{171}$$

$T_i = i\eta$ gives $\sum_{i=0}^{k-1} \eta^2 \mathsf{K}(T_k - T_i) = \eta^2 \sum_{m=1}^{k} \mathsf{K}(m\eta)$. Decompose $\mathsf{K}(z) = \sum_{j=1}^{r} K_j^{top}(z) + K^{tail}(z)$, where for $j \leq r$,

$$K_j^{top}(z) := a_{\text{top}}^2 \lambda_j^2 e^{-2a_{\text{top}}\lambda_j z}, \qquad K^{tail}(z) := a_{\text{full}}^2 (R^{\nu} + a_{\text{full}} z)^{-p}.$$

Let

$$J_{\mathsf{K}}(t) := \int_0^t \mathsf{K}(z)\,\mathrm{d}z = \frac{a_{\text{top}}}{2} \sum_{j=1}^{r} \lambda_j \left(1 - e^{-2a_{\text{top}}\lambda_j t}\right) + \frac{a_{\text{full}}}{1 - \nu^{-1}} \left[R^{1-\nu} - (R^{\nu} + a_{\text{full}} t)^{-(1-\nu^{-1})} \right]. \tag{172}$$

For $1 \leq m \leq k$,

$$\begin{aligned} \frac{K_j^{top}(m\eta)}{K_j^{top}((m-1)\eta)} &= e^{-2a_{\text{top}}\lambda_j \eta} \geq e^{-2a_{\text{top}}\lambda_1 \eta}, \\ \frac{K^{tail}(m\eta)}{K^{tail}((m-1)\eta)} &= \left(1 + \frac{a_{\text{full}}\eta}{R^{\nu} + a_{\text{full}}(m-1)\eta} \right)^{-p} \geq \left(1 + \frac{a_{\text{full}}\eta}{R^{\nu}} \right)^{-p}. \end{aligned} \tag{173}$$

Set the constant $c_m := \min\left\{ e^{-2a_{\text{top}}\lambda_1\eta}, \left(1 + \frac{a_{\text{full}}\eta}{R^{\nu}}\right)^{-p} \right\}$. Then (173) implies that for $z \in [(m-1)\eta, m\eta]$, $\mathsf{K}(m\eta) \leq \mathsf{K}(z) \leq \mathsf{K}((m-1)\eta) \leq c_m^{-1}\mathsf{K}(m\eta)$. Thus

$$c_m \int_{(m-1)\eta}^{m\eta} \mathsf{K}(z)\,\mathrm{d}z \leq \eta \mathsf{K}(m\eta) \leq \int_{(m-1)\eta}^{m\eta} \mathsf{K}(z)\,\mathrm{d}z.$$

It follows that

$$c_m \eta J_{\mathsf{K}}(t_k) \leq \eta^2 \sum_{m=1}^{k} \mathsf{K}(m\eta) \leq \eta J_{\mathsf{K}}(t_k). \tag{174}$$

By the hyper-parameter constraints in Theorem 7.2, we have

$$2a_{\text{top}}\lambda_1\eta \leq 2a_{\text{full}}\lambda_1\eta \leq \frac{\alpha_{\text{slow}}}{4} \leq \frac{1}{4}, \qquad \frac{a_{\text{full}}\eta}{R^{\nu}} = \frac{a_{\text{full}}\lambda_1\eta}{\lambda_1 R^{\nu}} \leq \frac{\alpha_{\text{slow}}}{8\lambda_1 R^{\nu}} \leq \frac{1}{8\ell_{-}R^{\nu}} \leq \frac{1}{8\ell_{-}}. \tag{175}$$

Consequently, $c_m \geq c_{m,0} := \min\left\{ e^{-1/4}, \left(1 + \frac{1}{8\ell_{-}}\right)^{-p} \right\} \gtrsim 1$.

Since $d \geq 2R$, we have

$$\sum_{j=R}^{d} \lambda_j \geq \sum_{j=R}^{2R-1} \ell_- j^{-\nu} \geq \ell_- 2^{-\nu} R^{1-\nu}, \qquad \sum_{j=R}^{d} \lambda_j \leq \ell_+ \sum_{j=R}^{\infty} j^{-\nu} \leq \frac{\ell_+}{q} R^{1-\nu}. \tag{176}$$

Then defining $c_{tr} = \min\left\{\frac{1}{2}, \frac{1}{\ell_+}\right\}$, $C_{tr} = \max\left\{\frac{1}{2}, \frac{2^\nu}{(1-\nu^{-1})\ell_-}\right\}$ and using $\operatorname{tr}(AH) = a_{\text{top}} \sum_{j=1}^{r} \lambda_j + a_{\text{full}} \sum_{j=R}^{d} \lambda_j$, we get

$$c_{tr} \operatorname{tr}(AH) \leq = \underbrace{\frac{a_{\text{top}}}{2} \sum_{j=1}^{r} \lambda_j + \frac{a_{\text{full}}}{1-\nu^{-1}} R^{1-\nu}}_{= J_{\mathsf{K}}(\infty)} \leq C_{tr} \operatorname{tr}(AH). \tag{177}$$

The constraint (169) implies that $a_{\text{top}} \lambda_j t_k \geq \ell_- C_h \left(\frac{r}{j}\right)^\nu (1 + \log \kappa) \geq \ell_- C_h$. Combining this with (171) implies that for every $j \leq r$,

$$1 - e^{-2a_{\text{top}} \lambda_j t_k} \geq 1 - e^{-2\ell_- C_h} \geq \frac{1}{2}. \tag{178}$$

Moreover, $\kappa \geq 1$ and $r/R = r/(r+1) \geq 4/5$ give

$$\frac{a_{\text{full}} t_k}{R^\nu} \geq \kappa C_h \left(\frac{r}{R}\right)^\nu (1 + \log \kappa) \geq C_h \left(\frac{4}{5}\right)^\nu. \tag{179}$$

This implies that

$$\frac{R^{1-\nu} - (R^\nu + a_{\text{full}} t_k)^{-(1-\nu^{-1})}}{R^{1-\nu}} = 1 - \left(1 + \frac{a_{\text{full}} t_k}{R^\nu}\right)^{-(1-\nu^{-1})} \geq 1 - \left[1 + C_h \left(\frac{4}{5}\right)^\nu\right]^{-(1-\nu^{-1})} \geq \frac{1}{2}. \tag{180}$$

Applying (178) and (180) to the closed form of $J_{\mathsf{K}}(t_k)$ yields

$$\frac{1}{2} J_{\mathsf{K}}(\infty) \leq J_{\mathsf{K}}(t_k) \leq J_{\mathsf{K}}(\infty). \tag{181}$$

Combining (174), (177), and (181) gives

$$\frac{c_{m,0} c_{tr}}{2} \eta \operatorname{tr}(AH) \leq \eta^2 \sum_{m=1}^{k} \mathsf{K}(m\eta) \leq C_{tr} \eta \operatorname{tr}(AH). \tag{182}$$

Part 2: estimate the signal term. By (83), there exists a fixed $C_{top} > 0$ such that the lower inequality in (169) implies

$$\sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\text{top}} \lambda_j t_k} \leq C_{top} (a_{\text{full}} t_k)^{-s}. \tag{183}$$

(171) and (179) imply $a_{\text{full}} t_k \geq R^\nu$, and therefore

$$2^{-s} (a_{\text{full}} t_k)^{-s} \leq (R^\nu + a_{\text{full}} t_k)^{-s} \leq (a_{\text{full}} t_k)^{-s}. \tag{184}$$

Using (183) and (184) to $\mathsf{S}(t) = \sum_{j=1}^{r} \lambda_j |w_j^\star|^2 e^{-2a_{\text{top}} \lambda_j t} + \left((r+1)^\nu + a_{\text{full}} t\right)^{-s}$, we have

$$2^{-s} (a_{\text{full}} \eta k)^{-s} \leq \mathsf{S}(\eta k) \leq (C_{top} + 1)(a_{\text{full}} \eta k)^{-s}. \tag{185}$$

Finally, combining (182) and (185) gives (170). □

Corollary I.16 (Acceleration by Slow Momentum)**.** *Suppose the conditions of Corollary I.14 hold. Define the following quantities as functions of χ:*

$$a_\chi = \gamma + \frac{1}{\alpha_{\text{fast}}} + \frac{\chi}{\alpha_{\text{slow}}},\ \kappa_\chi = \frac{a_\chi}{a_0},\ N_\chi = a_0(\Lambda_P + \kappa_\chi \Lambda_Q),\ \Gamma_\chi = \frac{\Lambda_P + \Lambda_Q}{\Lambda_Q + \Lambda_P/\kappa_\chi}, \tag{186}$$

where $\Lambda_P = \sum_{j=1}^{r} j^{-\nu}$, $\Lambda_Q = \sum_{j=r+1}^{d} j^{-\nu}$. Then the following statements hold:

1. ***(Proxy-optimal loss reduction).*** *The optimal learning rate* $\eta^\star_{\chi,k} := \operatorname{argmin}_\eta \mathcal{E}_\chi(\eta; k)$ *satisfies*

$$\eta^\star_{\chi,k} \asymp \left[\frac{s}{\sigma^2 a_0 N_\chi (a_\chi k)^s}\right]^{\frac{1}{s+1}}, \ \mathcal{E}_\chi(\eta^\star_{\chi,k}; k) \asymp \left[\frac{\sigma^2}{k}\left(\Lambda_Q + \frac{\Lambda_P}{\kappa_\chi}\right)\right]^{\frac{s}{s+1}}, \tag{187}$$

where the signal learning term, noise accumulation term, and risk $\mathcal{E}_\chi(\eta^\star_{\chi,k}; k)$ *are balanced at the same asymptotic order (up to constants). Consequently, the loss reduction ratio is* $\frac{\mathcal{E}_\chi(\eta^\star_{\chi,k};k)}{\mathcal{E}_0(\eta^\star_{0,k};k)} \asymp \Gamma_\chi^{-s/(s+1)}$.

2. ***(Iteration complexity speedup).*** *For any* $\epsilon > 0$*, let* $k_\chi(\epsilon)$ *solve* $\mathcal{E}_\chi(\eta^\star_{\chi,k}; k_\chi(\epsilon)) = \epsilon$*. Then we have*

$$\frac{k_\chi(\epsilon)}{k_0(\epsilon)} \asymp \Gamma_\chi^{-1}. \tag{188}$$

The proof follows by direct computation with using the weighted AM-GM Inequality to (170).

I.3 Useful Lemmas

Theorem I.17 (Gershgorin disk theorem)**.** *For* $M = (m_{ij}) \in \mathbb{C}^{n\times n}$*, define the row disks*

$$D_i := \left\{ z \in \mathbb{C} : |z - m_{ii}| \le \sum_{j\neq i} |m_{ij}| \right\}.$$

Every eigenvalue of M *belongs to* $\bigcup_{i=1}^n D_i$*. More precisely, if a connected component of this union consists of exactly* k *disks and is disjoint from all remaining disks, then it contains exactly* k *eigenvalues of* M*, counted with algebraic multiplicity.*

Theorem I.18 (Neumann series and matrix resolvent identities)**.** *Let* $E \in \mathbb{C}^{n\times n}$*, and the norm* $\|\cdot\| \in \{\|\cdot\|_2, \|\cdot\|_\infty\}$*. If* $\|E\| < 1$*, then*

$$(I - E)^{-1} = \sum_{n=0}^{\infty} E^n, \qquad \|(I-E)^{-1}\| \le \frac{1}{1 - \|E\|}. \tag{189}$$

For $A \in \mathbb{C}^{n\times n}$*, write* $R_A(z) := (zI - A)^{-1}$ *whenever* $zI - A$ *is invertible. If* $\|R_A(z)E\| < 1$*, then*

$$R_{A+E}(z) = (I - R_A(z)E)^{-1} R_A(z), \qquad \|R_{A+E}(z)\| \le \frac{\|R_A(z)\|}{1 - \|R_A(z)E\|}. \tag{190}$$

Whenever $zI - A$ *and* $wI - A$ *are invertible,*

$$R_A(z) - R_A(w) = (w - z) R_A(z) R_A(w). \tag{191}$$

For $A, B \in \mathbb{C}^{n\times n}$*, whenever both* $zI - A$ *and* $zI - B$ *are invertible,*

$$R_A(z) - R_B(z) = R_A(z)(A - B)R_B(z). \tag{192}$$

Theorem I.19 (Riesz spectral projection formula)**.** *Let* $A \in \mathbb{C}^{n\times n}$*, and* Γ *be a positively oriented, smooth, simple closed contour with no eigenvalue of* A *on the contour. The matrix*

$$P_\Gamma(A) := \frac{1}{2\pi \mathrm{i}} \int_\Gamma (zI - A)^{-1} \, \mathrm{d}z \tag{193}$$

is the projection (i.e., $P_\Gamma(A)^2 = P_\Gamma(A)$*) onto the generalized eigenspaces corresponding to the eigenvalues inside* Γ*. It commutes with* A*, and for every integer* $n \ge 0$*,*

$$A^n P_\Gamma(A) = \frac{1}{2\pi \mathrm{i}} \int_\Gamma z^n (zI - A)^{-1} \, \mathrm{d}z. \tag{194}$$

If no eigenvalue of $B \in \mathbb{C}^{n\times n}$ *lies on* Γ*, then*

$$P_\Gamma(A) - P_\Gamma(B) = \frac{1}{2\pi \mathrm{i}} \int_\Gamma R_A(z)(A - B)R_B(z) \, \mathrm{d}z. \tag{195}$$