BEYOND THE MATRIX SIGN: QUADRATIC SPECTRAL DESCENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Muon emerges as a strong competitor of the AdamW for LLM pretraining, because the matrix-wise update it employs can potentially incur smaller second-order penalty than the once dominating AdamW, which performs coordinate-wise update. However, the spectral flattening procedure in Muon is quite debatable since it discards the spectral amplitude information totally. To seek for better spectral allocation (and the associated spectral subspace), we propose to solve the quadratic model of loss function under the spectral norm constraint *directly* (i.e., in a genuinely Newtonian way) and thus obtaining the Quadratic Spectral Descent (QSD) algorithm. In contrast, many existing curvature-aware methods either exploit the second-order information in an *implicit* way by changing the weight update geometry (such as Mousse, FISMO) or rely on strong assumptions (such as the weight displacement isotropy assumption in Newton-Muon). QSD's potential advantage over these methods is best illustrated in the isotropic curvature scenario, where Mousse, FISMO and Newton-Muon all reduce to Muon while the spectral allocation in QSD is still *non-flat* (since the spectral allocation in QSD depends on the *gradient to curvature ratio*). Meanwhile, to control the complexity of QSD, we employ inversion-free K-FAC and *online* Frank-Wolfe update which is essentially a matrix sign operator. Overall, the complexity increase can be rather mild. Experiments on GPT pre-training show that QSD consistently improves validation loss over Muon and recent Muon variants, while achieving up to an 8.49% wall-clock speedup at matched validation loss.

1 INTRODUCTION

AdamW (Loshchilov & Hutter, 2017) is the de-facto standard optimizer for large-scale neural networks. However, its role has been challenged recently by Muon (Jordan et al., 2024b; Liu et al., 2025; Shah et al., 2025; Wen et al., 2026), which distinguishes itself as a matrix-valued optimizer with spectral norm constraint, as opposed to the conventional vector-valued optimizers constrained by the Euclidean norm. By employing the matrix-wise update (specifically, matrix orthogonalization and spectral flattening), Muon has the potential advantage over AdamW which uses coordinate-wise update in that: Muon can incur less second-order penalty than AdamW while the first-order gain of them two are similar, as reported in Wang et al. (2026). Thus Muon can achieve larger loss decrease per step compared with AdamW overall.

Muon can be proved to be the best optimizer for the *linear* model under the spectral norm constraint. In specific, in an idealized form, Muon computes (Pethick et al., 2025; Bernstein & Newhouse, 2024)

$$D_{\mathrm{G}} = -\rho \ \mathrm{msgn}(G) = -\rho UV^\top = \arg\min_{\|D\|_2 \le \rho} \langle G, D \rangle, \tag{1}$$

where $G = U\Sigma V^\top$. Meanwhile, Muon indeed demonstrates superior performance than AdamW in practical LLM pretraining (Liu et al., 2025).

Despite of the above, one of the key operation of Muon, i.e. spectral flattening, is still quite debatable even questionable, since by that operation all the spectrum amplitudes information is discarded. A natural question is thus: *can we find a more efficient way for spectral allocation (as well as the associated spectral subspace update), if we are provided some kind of second-order information?*

The source code is publicly available at this link.

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.

Table 1: Comparison of spectral matrix optimizers. Here, $\mathrm{Polar}(X) = UV^\top$ for the singular value decomposition $X = U\Sigma V^\top$.

Method	Optimization Objective	Role of Curvature	Update form
Muon	$\min_{\|D\|_2 \le \rho} \langle G, D\rangle$	None	$D = -\rho\ \mathrm{Polar}(G)$
Newton–Muon	$\min_{Q} -\langle G, Q\rangle + \frac{1}{2}\mathrm{tr}(HQAQ^\top)$	Partial (A factor only $\rightarrow$ right preconditioning)	$D \propto -\mathrm{Polar}(GA^{-1})$
Mousse	$\min_{D}\langle G, D\rangle \quad \text{s.t.} \quad \|L^{1/4}DR^{1/4}\|_2 \le \rho$	Implicit (Shampoo-style preconditioning)	$D \propto -L^{-1/4}\mathrm{Polar}(L^{-1/4}GR^{-1/4})R^{-1/4}$
FISMO	$\min_{D}\langle G, D\rangle \quad \text{s.t.} \quad \|P^{1/2}DQ^{1/2}\|_2 \le \rho$	Implicit (Fisher preconditioning)	$D = -\rho P^{-1/2}\mathrm{Polar}(P^{-1/2}GQ^{-1/2})Q^{-1/2}$
QSD	$\min_{\|D\|_2 \le \rho}\left\{\langle G, D\rangle + \frac{\eta}{2}\mathrm{vec}(D)^\top C\ \mathrm{vec}(D)\right\}$	Explicit (in the objective)	Frank-Wolfe Iterates

To answer this question, several curvature-aware Muon variants have been proposed, including Newton-Muon (Du & Su, 2026), Mousse (Zhang et al., 2026b), FISMO (Xu et al., 2026) etc. Newton-Muon (Du & Su, 2026) is based on an important yet strong assumption, i.e., the weight displacement isotropy, thus it can remove the effect of the output curvature and finally obtain a right-preconditioned matrix sign solver. Mousse (Zhang et al., 2026b) employs the Shampoo-style second-order information (Gupta et al., 2018; Morwani et al., 2025)(namely gradient covariance) to whiten the weight update. FISMO (Xu et al., 2026) instead employs the K-FAC-based (Martens & Grosse, 2015) Fisher information matrix (FIM) to modify the local geometry of the weight update (i.e., natural gradient trust region) . All these second-order Muon variants are reported to outperform the vanilla Muon. However, it can be seen that all of them either rely on strong assumption (Newton-Muon) or exploit the second-order information in an *implicit* way (Mousse, FISMO) by changing the local weight geometry and thus resulting in a preconditioned matrix sign solution. When the curvature is isotropic, unfortunately, it turns out that all the above Muon variants will reduce to Muon, i.e., the spectrum will still be flat, thus at the risk of performance degradation (Su, 2025).

To fully tackle the above question, in this paper we will take a different yet more direct route. Specifically, we will solve the quadratic model under the spectral-norm constraint *directly* in the *general* form (i.e., without any assumptions), thus obtaining the Quadratic Spectral Descent (QSD) algorithm. Concretely, QSD aims to solve:

$$\min_{\|D\|_2 \le \rho} \left\{\langle G, D\rangle + \frac{\eta}{2}\mathrm{vec}(D)^\top C\ \mathrm{vec}(D)\right\}, \tag{2}$$

where $C \succeq 0$ models the local curvature. It is worth noting that for this quadratic objective optimization, the solution is $s_i^\star = \min\left\{\rho, \frac{g_i}{\eta\kappa}\right\}$ in the isotropic-curvature and aligned-spectrum (i.e., gradient and curvature) case, where $s_i^\star$ and g_i denote the spectral allocation and gradient strength of the i-th mode, respectively, and κ is the isotropic curvature, implying non-flat spectral allocation generally. This is in stark contrast to the above-discussed second-order Muon variants. The differences among them are summarized in Table 1.

Furthermore, to make the quadratic spectral problem practical at modern training scale, QSD employs K-FAC (Martens & Grosse, 2015; George et al., 2018) only to obtain structured curvature with no matrix inversion involved. Then we solve the constrained quadratic by *online* Frank-Wolfe updates (Frank & Wolfe, 1956; Jaggi, 2013). The total complexity increase of QSD relative to Muon turns out to be rather mild due to two important reasons: 1) Online optimization: we employ the warm-start over each outer iteration, thus offering a high-quality initial value for the inner Frank-Wolfe updates; 2) Closed-form Frank-Wolfe solution: each Frank-Wolfe step has a closed-form solution, namely the matrix sign operator as in Muon. Together, they make it possible that 3 Frank-Wolfe steps for each outer iteration are sufficient, thus substantially lowering the total complexity increase.

Overall, our analysis shows that the quadratic model changes the solution of linear model for Muon significantly, in terms of both spectral amplitude and spectral directions. Specifically, when the gradient and curvature are aligned in terms of singular directions, the spectral amplitudes should be proportional to the *gradient-to-curvature ratio*, rather than being uniform (as in Muon). Moreover, if they are not aligned, the optimal singular directions also need rotation relative to those of Muon. Experiments on language-model pretraining show that QSD achieves these curvature-aware updates with mild computation overhead and consistent performance improvement over Muon.

Our contributions are summarized as follows:

- **Quadratic spectral optimization.** We formulate QSD by directly optimizing a quadratic local model under the same spectral-norm constraint as in Muon. We theoretically and empirically show that curvature generally induces a nonuniform spectral allocation and can also rotate the preferred singular subspace. Notably, even under isotropic curvature, QSD remains non-flat while several curvature-aware Muon variants reduce to Muon.
- **Scalable curvature-aware solver.** We develop an inversion-free implementation based on Kronecker-factored curvature and a few online Frank-Wolfe steps. Each inner linear oracle has the same matrix-sign form as Muon, while momentum and curvature smoothing enable effective warm starts.
- **Optimization guarantees and empirical gains.** We derive a computable Frank-Wolfe optimality certificate, show that curvature-aware refinement can improve upon the Muon update under the same quadratic surrogate, and establish an $O(1/K)$ convergence rate for the inner solver. On GPT pre-training, QSD consistently improves validation loss over Muon and recent Muon variants, and reduces wall-clock training time by up to 8.49% at matched validation loss.

1.1 Related Work

Muon and spectral-norm optimization. Muon can be interpreted as normalized steepest descent under the spectral norm, or equivalently as linear optimization over a spectral-norm ball (Bernstein & Newhouse, 2024; Crawshaw et al., 2025; Riabinin et al., 2025; Pethick et al., 2025). This viewpoint explains the matrix-sign update as the exact solution of a linear minimization oracle over the spectral-norm ball and connects Muon to more general Linear Minimization Oracle (LMO)-based matrix optimization methods such as Scion (Pethick et al., 2025). Related analyses have studied the geometry, convergence, and implicit properties of Muon-style updates. Our work starts from the same spectral-norm geometry, but asks what update is preferred when the local objective is extended from linear to quadratic.

Curvature-aware and matrix optimization methods. Recent work has incorporated curvature or preconditioning into Muon-style optimization. Newton-Muon, FISMO, Mousse, and GO-MUON use activation, Fisher, Shampoo, or K-FAC-style statistics to transform the gradient or define an anisotropic spectral geometry before applying a spectral or polar update (Du & Su, 2026; Xu et al., 2026; Zhang et al., 2026b). A complementary line directly modifies the singular spectrum of Muon updates through adaptive or nonuniform spectral transformations (Dong & Sawin, 2026; Wu et al., 2026b;a; Pang et al., 2026). These methods are related to structured preconditioners such as K-FAC, Shampoo, and SOAP (Martens & Grosse, 2015; George et al., 2018; Gupta et al., 2018; Morwani et al., 2025; Vyas et al., 2025), which reduce the cost of full-matrix curvature but still rely on matrix inversions, inverse roots, or eigendecompositions to construct the preconditioner. QSD takes a different route: it keeps the spectral-norm constraint fixed and uses curvature directly in the quadratic objective. It therefore avoids curvature inversion while jointly optimizing the singular directions and singular values of the finite update.

2 Preliminaries and Problem Formulation

We consider the optimization of a matrix-valued parameter $W \in \mathbb{R}^{m\times n}$ with objective $f(W)$. Let $G = \nabla_W f(W)$ denote the gradient with respect to W. We write a single parameter update as $W^+ = W + \eta D$, where $D \in \mathbb{R}^{m\times n}$ is the update direction and $\eta > 0$ is the learning rate. Throughout the paper, $\|\cdot\|_2$ denotes the spectral norm, $\|\cdot\|_*$ the nuclear norm, $\|\cdot\|_F$ the Frobenius norm, and $\langle X, Y\rangle = \mathrm{tr}(X^\top Y)$ the Frobenius inner product. Building on the constrained interpretation of Muon in Eq. 1, we extend the same spectral-norm geometry from a linear to a quadratic local model as follows:

For a finite update, directions with similar first-order descent can incur substantially different second-order costs. Let $H_W = \nabla^2_{\mathrm{vec}(W)} f(W)$ denote the Hessian and define $\mathcal{H}[D] = \mathrm{unvec}\,(H_W\,\mathrm{vec}(D))$. A second-order Taylor expansion gives

$$f(W + \eta D) = f(W) + \eta\langle G, D\rangle + \frac{\eta^2}{2}\langle D, \mathcal{H}[D]\rangle + \mathcal{R}_3(D). \tag{3}$$

As the Hessian is expensive to apply, we therefore use a positive-semidefinite curvature operator $\mathcal{C}$ instead. After removing the constant term and dividing by η, we obtain

$$q_\eta(D) = \langle G, D\rangle + \frac{\eta}{2}\langle D, \mathcal{C}[D]\rangle. \tag{4}$$

We define Quadratic Spectral Descent (QSD) as

$$\boxed{D^\star \in \arg\min_{\|D\|_2 \le \rho} q_\eta(D).} \tag{5}$$

When $\mathcal{C} = 0$, QSD reduces exactly to the Muon problem.

3 Practical Quadratic Spectral Descent

The QSD problem is only useful if its curvature term can be estimated and the constrained quadratic can be solved efficiently. We address these two issues with K-FAC and Frank–Wolfe, respectively.

3.1 Kronecker-Factored Curvature

Consider a linear layer $h = Wa$, where $W \in \mathbb{R}^{m\times n}$ and $a \in \mathbb{R}^n$. Let $\delta = \nabla_h \ell \in \mathbb{R}^m$. For a single example, the gradient with respect to W has the outer-product form $\nabla_W \ell = \delta a^\top$. The corresponding empirical-Fisher block is $C_{\mathrm{EF}} = \mathbb{E}\left[(aa^\top) \otimes (\delta\delta^\top)\right]$. We use the empirical-Fisher factors as an inexpensive positive-semidefinite structured curvature proxy, without assuming that the empirical Fisher is an exact approximation to the GGN or Hessian (Schraudolph, 2002; Kunstner et al., 2019). Its overall scale is calibrated against the GGN directional curvature in Section 3.3. K-FAC approximates this block by factorizing the expectation (Martens & Grosse, 2015; George et al., 2018):

$$\widehat{C}_{\mathrm{KFAC}} = A \otimes B, \qquad A = \mathbb{E}[aa^\top], \qquad B = \mathbb{E}[\delta\delta^\top]. \tag{6}$$

Here A captures input-side second-order statistics and B captures output-gradient statistics. The development below only requires $A \succeq 0$ and $B \succeq 0$, so analogous factorizations can also be used for other positive-semidefinite curvature models. Using $\mathrm{vec}(BDA) = (A^\top \otimes B)\,\mathrm{vec}(D)$, and the symmetry of A, the corresponding matrix-valued curvature action is $\widehat{\mathcal{C}}[D] = BDA$.

Importantly, K-FAC is used here to define a local quadratic *objective*, rather than merely to construct a preconditioned gradient. We optimize this curvature-aware objective directly over the spectral-norm ball.

3.2 Frank–Wolfe Solver

The QSD problem does not in general have a closed-form solution. We therefore use Frank-Wolfe (Frank & Wolfe, 1956; Jaggi, 2013) to solve it approximately. At each step, Frank-Wolfe linearizes the quadratic objective at the current iterate, solves the resulting linear problem over the same spectral-norm ball, and moves toward its solution. Although the original problem is quadratic, each linear subproblem still has a closed-form matrix-sign solution.

Let D_s denote the current inner iterate. The gradient of the quadratic surrogate with respect to D_s is

$$R_s = \nabla_D \widehat{q}_\eta(D_s) = G + \eta B D_s A. \tag{7}$$

The second term corrects the original gradient G according to the curvature at the current iterate. Frank-Wolfe then solves

$$S_s \in \arg\min_{\|S\|_2 \le \rho} \langle R_s, S\rangle, \qquad S_s = -\rho\ \mathrm{msgn}(R_s). \tag{8}$$

The new candidate is obtained by moving toward this feasible atom:

$$D_{s+1} = (1-\gamma_s)D_s + \gamma_s S_s, \qquad \gamma_s \in [0,1]. \tag{9}$$

For the quadratic surrogate, γ_s can be obtained by exact line search. Letting $\Delta_s = S_s - D_s$, we have

$$\gamma_s = \mathrm{clip}_{[0,1]}\left(\frac{\langle R_s, D_s - S_s\rangle}{\eta\ \mathrm{tr}\,(\Delta_s^\top B \Delta_s A)}\right), \tag{10}$$

With $D_0 = 0$, we have $R_0 = G$, and hence the first Frank-Wolfe atom is exactly the Muon direction. Subsequent atoms use the curvature-corrected gradient R_s and progressively refine this initial update.

Algorithm 1 Practical Quadratic Spectral Descent

Require: Momentum M_t, curvature operator $\mathcal{C}_t$, spectral radius ρ, FW steps K
1: $D_{t,0} \leftarrow D_{t-1,K}$ {use $D_{0,0} = 0$}
2: **for** $s = 0, \dots, K-1$ **do**
3: $\quad R_{t,s} \leftarrow M_t + \eta_t \, \mathcal{C}_t[D_{t,s}]$
4: $\quad S_{t,s} \leftarrow -\rho \, \mathrm{msgn}(R_{t,s})$
5: $\quad \Delta_{t,s} \leftarrow S_{t,s} - D_{t,s}$
6: $\quad \gamma_{t,s} \leftarrow \arg\min_{\gamma \in [0,1]} q_t(D_{t,s} + \gamma \Delta_{t,s})$
7: $\quad D_{t,s+1} \leftarrow D_{t,s} + \gamma_{t,s} \Delta_{t,s}$
8: **end for**
9: $W_{t+1} \leftarrow W_t + \eta_t D_{t,K}$

3.3 PRACTICAL QSD

The derivation above assumes fixed curvature factors. In practice, we estimate the K-FAC factors from minibatch statistics and update them periodically during training.

Factor estimation. At a factor-refresh iteration t, using a sample subset $\mathcal{S}_t$, we estimate for each layer ℓ

$$\widehat{A}_{t,\ell} = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} a_{i,\ell} a_{i,\ell}^\top, \qquad \widehat{B}_{t,\ell} = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \delta_{i,\ell} \delta_{i,\ell}^\top, \tag{11}$$

and maintain exponential moving averages

$$A_{t,\ell} = \beta_1 A_{t-j,\ell} + (1-\beta_1)\widehat{A}_{t,\ell}, \qquad B_{t,\ell} = \beta_1 B_{t-j,\ell} + (1-\beta_1)\widehat{B}_{t,\ell}, \tag{12}$$

where j is the factor-refresh interval. Between refreshes, the most recent factors are reused. Details could be found in Appendix H.

Curvature calibration. To mitigate the possible scale mismatch in the Kronecker approximation, we introduce a per-layer calibration coefficient $\alpha_{t,\ell} > 0$ by matching the K-FAC and generalized Gauss–Newton (GGN) directional curvatures along the recent update direction. To reduce noise, the ratio is smoothed by:

$$\alpha_{t,\ell} = \beta_\alpha \alpha_{t-k,\ell} + (1-\beta_\alpha) \, \mathrm{clip}\left(\widehat{\alpha}_{t,\ell}, \alpha_{\min}, \alpha_{\max}\right). \tag{13}$$

Full details are given in Appendix I.

Even after calibration, the quadratic surrogate can still differ from the practical finite-step training objective. We therefore apply a mild curvature inflation factor $\tau > 1$ to make the update more conservative in directions where curvature already plays a significant role. The effect of this inflation is analyzed in Appendix F. We use $\tau = 1.5$ throughout our experiments.

Together with isotropic damping, the practical curvature operator is

$$\widehat{\mathcal{C}}_{t,\ell}[D] = \tau(\alpha_{t,\ell} B_{t,\ell} D A_{t,\ell} + dD), \tag{14}$$

where $d \geq 0$ is the damping coefficient.

Practical Frank-Wolfe update. For clarity, we suppress the layer index ℓ below. As in Muon, we use the momentum-smoothed matrix M_t in place of the instantaneous gradient. The surrogate gradient at the s-th Frank-Wolfe iterate is therefore

$$R_{t,s} = M_t + \eta_t \left(\tau(\alpha_t B_t D_{t,s} A_t + d D_{t,s})\right). \tag{15}$$

Since consecutive inner problems change gradually, we warm-start the solver from the previous solution, $D_{t,0} = D_{t-1,K}$, with $D_{0,0} = 0$. The warm start remains feasible and allows a small number of Frank–Wolfe steps to track the evolving QSD solution. Algorithm 1 summarizes the resulting practical update.

Table 2: Main computational costs for a matrix parameter $W \in \mathbb{R}^{m\times n}$. Here s is the number of samples used to estimate curvature, j is the curvature-factor update interval, and K is the number of Frank–Wolfe steps.

Cost	Dense curvature	Muon	QSD
Curvature storage	$\mathcal{O}(m^2n^2)$	–	$\mathcal{O}(m^2+n^2)$
Curvature estimation	$\mathcal{O}(sm^2n^2)$	–	$\mathcal{O}\big(s(m^2+n^2)/j\big)$
Curvature application	$\mathcal{O}(m^2n^2)$	–	$\mathcal{O}(m^2n+mn^2)$ per FW step
Factorization / inversion	$\mathcal{O}(m^3n^3)$	–	Not required
Matrix-sign operation	–	$1\times$	$K\times$

3.4 Optimization guarantees.

The Frank-Wolfe gap provides a computable certificate of suboptimality for the QSD inner problem. With exact line search, the surrogate objective decreases monotonically; moreover, under zero initialization, the first Frank-Wolfe step is no worse than the Muon update under the same quadratic surrogate. The inner solver converges to the optimal surrogate value at the standard $O(1/K)$ Frank-Wolfe rate. Formal statements and proofs are provided in Appendix B.

3.5 Computational Complexity

Despite using curvature information and an iterative inner solver, QSD incurs only a 6% wall-clock overhead over Muon in our 124M model experiments. This modest overhead is enabled by structured curvature, infrequent factor updates, and warm-started inner solves.

For $W \in \mathbb{R}^{m\times n}$, the full curvature acts on an mn-dimensional space, requiring $\mathcal{O}(m^2n^2)$ storage and application cost, and up to $\mathcal{O}(m^3n^3)$ for direct factorization or inversion. QSD instead uses the K-FAC structure $\mathcal{C}_t[D] = B_tDA_t$, where $A_t \in \mathbb{R}^{n\times n}$ and $B_t \in \mathbb{R}^{m\times m}$. This reduces curvature storage to $\mathcal{O}(m^2+n^2)$ and each curvature application to $\mathcal{O}(m^2n+mn^2)$, without requiring curvature inversion.

The factors are refreshed only periodically and reused between optimizer steps. If they are updated every j steps using s samples, their amortized estimation cost is $\mathcal{O}\left(s(m^2+n^2)/j\right)$. The curvature-scale calibration is performed only every roughly 200 or more optimizer steps, uses only a small fraction of the training data, and requires only additional forward evaluations without an extra backward pass. Its amortized contribution to the overall training cost is therefore small.

QSD solves the quadratic spectral subproblem using a small number K of Frank-Wolfe steps, each requiring one structured curvature application and one matrix-sign operation. For approximately square matrices, $m \approx n \approx d$, this gives an additional optimizer cost of $\mathcal{O}(Kd^3)$. We further warm start with the previous solution, $D_{t,0} = D_{t-1,K}$, since consecutive subproblems change gradually as the momentum evolves and the curvature factors are updated only periodically. This allows only a few online Frank-Wolfe steps to suffice in practice.

For comparison, the main forward and backward computation of a $d \times d$ layer over N token representations scales as $\mathcal{O}(Nd^2)$, whereas the QSD inner solve scales as $\mathcal{O}(Kd^3)$ and does not grow with the number of processed tokens. Complete wall-clock comparisons are reported in Section 5.

4 Understanding How Curvature Changes Muon Updates

We now characterize how the quadratic curvature term changes the structure of the optimal update relative to Muon. We isolate two effects. First, when the gradient and curvature share the same singular modes, curvature alone is sufficient to produce nonuniform singular amplitudes. Second, under anisotropic curvature, the optimal update can also change its singular directions, even when the spectral-norm constraint is active. These results show that Muon's flat spectrum and its gradient-aligned singular directions are imposed by the spectral-norm constraint and linear optimization.

4.1 Curvature Induces Nonuniform Singular Amplitudes

We first isolate the effect of curvature on singular amplitudes. Suppose the singular modes of the gradient are also eigenmodes of the curvature operator. Let

$$G = U \operatorname{diag}(g_1, \ldots, g_p) V^\top = \sum_{i=1}^{p} g_i u_i v_i^\top, \qquad g_i \geq 0, \tag{16}$$

and define $E_i = u_i v_i^\top$. Assume $\mathcal{C}[E_i] = \kappa_i E_i$ with $\kappa_i \geq 0$. Under this aligned setting, curvature does not change the singular directions, but it can change the preferred magnitude of each mode.

Proposition 4.1 (Optimal spectrum under aligned curvature)**.** *Consider*

$$\min_{\|D\|_2 \leq \rho} \left\{ \langle G, D \rangle + \frac{\eta}{2} \langle D, \mathcal{C}[D] \rangle \right\}, \tag{17}$$

under aligned curvature. Then there exists an optimal solution of the form

$$D^\star = -U \operatorname{diag}\left(s_1^\star, \ldots, s_p^\star\right) V^\top, \tag{18}$$

where

$$s_i^\star = \begin{cases} \rho, & \kappa_i = 0, \\ \min\left\{ \rho, \dfrac{g_i}{\eta \kappa_i} \right\}, & \kappa_i > 0. \end{cases} \tag{19}$$

The proof is given in Appendix C.1.

Muon sets every active singular value to ρ. In contrast, Proposition 4.1 shows that the curvature-aware optimum saturates mode i only when $g_i \geq \eta \rho \kappa_i$. Otherwise, $s_i^\star = \frac{g_i}{\eta \kappa_i} < \rho$. Thus, even when gradient and curvature share the same singular directions, the quadratic optimum generally has a nonuniform singular-value profile. Intuitively, directions with larger curvature are more costly to move along, so they should receive smaller updates unless the gradient signal is sufficiently strong.

4.2 Curvature Can Change the Singular Directions

We next ask whether curvature can also change the singular directions of the optimal update. Under K-FAC curvature $\mathcal{C}[D] = BDA$ with $A \succ 0$ and $B \succ 0$, the unconstrained quadratic minimizer is $D_{\mathrm{unc}} = -\frac{1}{\eta} B^{-1} G A^{-1}$. For a rank-one gradient $G = g\, uv^\top$, this becomes $D_{\mathrm{unc}} = -\frac{g}{\eta}(B^{-1}u)(A^{-1}v)^\top$, which is generally not aligned with the singular directions of G. Thus anisotropic curvature naturally induces singular-direction adaptation.

One might ask whether this effect disappears when the spectral-norm constraint is active. The following result shows that it does not.

Proposition 4.2 (Direction adaptation with an active spectral constraint)**.** *Suppose*

$$\mathcal{C}[D] = DA, \qquad A \succ 0, \qquad G = g\, uv^\top,$$

where $g > 0$ and $\|u\|_2 = \|v\|_2 = 1$. If the unconstrained minimizer lies outside the spectral-norm ball, then there exists a unique $\lambda > 0$ such that

$$D^\star = -g\, u \left[(\eta A + \lambda I)^{-1} v \right]^\top, \qquad \|D^\star\|_2 = \rho. \tag{20}$$

If v is not an eigenvector of A, the right singular direction of $D^\star$ is not parallel to that of G.

The proof is given in Appendix C.2. The mechanism is easiest to see in the eigenbasis of the curvature. Let $Aq_j = a_j q_j$ and write $v = \sum_j c_j q_j$. Then $(\eta A + \lambda I)^{-1} v = \sum_j \frac{c_j}{\eta a_j + \lambda} q_j$.

Hence, different curvature components of v are rescaled by different amounts: high-curvature components are suppressed more strongly than low-curvature ones. Unless v is a curvature eigenvector, these unequal rescalings change its orientation. The multiplier $\lambda > 0$ adjusts the overall shrinkage so that the solution lies on the boundary $\|D^\star\|_2 = \rho$, but does not remove this anisotropic reweighting. Thus, the spectral-norm constraint controls the size of the update without forcing it to preserve the singular directions of the gradient.

A simple two-dimensional example makes this explicit. Let $B = I, A = \begin{pmatrix} 100 & 0 \\ 0 & 1 \end{pmatrix}, G = \begin{pmatrix} 101 & 2 \\ 0 & 0 \end{pmatrix}, \eta = 1$, and choose $\rho = \sqrt{2}$. The unconstrained quadratic minimizer gives

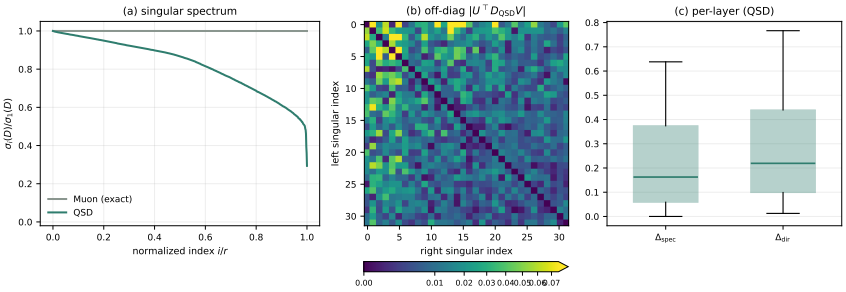


Figure 1: **QSD differs from Muon in both singular amplitudes and directions at a mild step of training.** **(a)** Normalized singular spectra of Muon and QSD for a representative layer. **(b)** Magnitude of $U^\top D_{\mathrm{QSD}} V$ in the momentum singular direction (top 32×32 block). Off-diagonal entries indicate changes in singular directions. **(c)** Distribution of Δ_{spec} and Δ_{dir} across QSD layers.

$D_{\mathrm{unc}} = -GA^{-1}$. Its spectral norm satisfies $\|D_{\mathrm{unc}}\|_2 > \sqrt{2} = \rho$, so the spectral-norm constraint is active. The KKT conditions in Proposition 4.2 are satisfied with $\lambda = 1$, which gives $D^\star = -G(A+I)^{-1} = -\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}, \|D^\star\|_2 = \sqrt{2} = \rho$. The right singular direction of G is proportional to $(101, 2)^\top$, whereas that of $D^\star$ is proportional to $(1, 1)^\top$. Thus, the optimal singular direction changes even though the solution lies on the boundary of the same spectral-norm ball used by Muon.

4.3 QSD vs. Muon in Practice

The preceding analysis shows that curvature can change both the singular amplitudes and singular directions of the update. We now examine whether these effects occur in practice. At a fixed training step, we compute the Muon and QSD directions from the same momentum matrix $M_t = U\Sigma V^\top$, thereby excluding differences caused by training trajectories. For this analysis, Muon is computed using the exact polar factor.

We measure spectral and directional deviations by

$$\Delta_{\mathrm{spec}}(D) = \frac{\|\sigma(D) - \bar{\sigma}(D)\mathbf{1}\|_2}{\|\sigma(D)\|_2}, \qquad \Delta_{\mathrm{dir}}(D; M) = \frac{\|D - \mathcal{P}_M(D)\|_F}{\|D\|_F}, \tag{21}$$

where $\bar{\sigma}(D)$ is the mean singular value and $\mathcal{P}_M(D) = U\,\mathrm{diag}(\mathrm{diag}(U^\top DV))V^\top$. Thus, Δ_{spec} measures deviation from a flat singular spectrum, while Δ_{dir} measures deviation from the singular directions of M. Both are zero for exact Muon.

Figure 1 shows that QSD differs from Muon in both aspects. Its singular spectrum is clearly nonuniform (Figure 1(a)), while $U^\top D_{\mathrm{QSD}} V$ contains substantial off-diagonal mass (Figure 1(b)), indicating that QSD also changes the singular directions. Across layers, both deviations remain consistently nonzero (Figure 1(c)). These observations confirm that the curvature-induced effects identified above also arise during training.

5 Experiments

5.1 Pre-training

Baseline. We first pre-train GPT-124M with a compute-optimal (Hoffmann et al., 2022) token-to-parameter (T2P) ratio. All models are trained on the FineWeb dataset (Penedo et al., 2024), with the same tokenizer as Muon Speed Run (Jordan et al., 2024a) and Newton-Muon (Du & Su, 2026), with a vocabulary size of 50257 and a batch size of 512. Training is conducted on RTX 5000 Ada GPUs using mixed precision (bfloat16). Following the setup in (Jordan et al., 2024a; Du & Su, 2026; Zhang et al., 2026a; Amsel et al., 2026), we apply QSD (or Muon) to all parameters except embeddings, unembeddings, normalization layers, and positional encodings, which are optimized using AdamW. For the polar operator, we adopt the same configuration as in (Jordan et al., 2024b). We sweep the AdamW learning rate in $[0.001, 0.0018, 0.0032, 0.0056, 0.01]$ and the remaining part in $[0.005, 0.01, 0.02, 0.04]$ and report the best results in Table 3. Detailed hyperparameters are provided

in the Appendix J. As shown in Table 3, QSD outperforms Muon at 6% time overhead. Due to the page limits, we defer the ablation for the baseline to Appendix G.

Table 3: Performance of pretraining GPT-124M on 4 RTX 5000 Ada GPUs.

Method	Run 1	Run 2	Run 3	Avg. loss	Avg. time (s)	Time Overhead (%)
Muon	3.3318	3.3312	3.3319	3.3316	7390.6	0.0
QSD	3.3151	3.3161	3.3161	**3.3158** (-0.0158)	7833.9	+6.00

GPT-350M. To verify whether the performance advantage of QSD can be maintained at a larger scale, we pre-trained GPT-350M under the optimal T2P ratio. We reuse the QSD-specific hyperparameters tuned on GPT-124M, and only adjust the learning rate together with the K-FAC factor-refresh and calibration intervals (and their EMA coefficients) to account for the longer training schedule. All other settings are kept unchanged; full values are listed in Table 14. As shown in Table 4, QSD can still ensure performance gains after the model scale is increased.

Table 4: Performance of pretraining GPT-350M on 4 RTX 5000 Ada GPUs.

Method	Val loss	Time (s)	Time Overhead (%)
Muon	3.0148	53569.3	0.0
QSD	**3.0018** (-0.013)	56986.3	+6.38

5.2 Training Efficiency

In this section, we report the token and the wall-clock time required to reach the same validation loss, where the target is set to the final validation loss of the QSD. The results are shown in Table 5. QSD speeds up pretraining by up to 8.49% in the evaluated settings.

Table 5: Training cost required to reach a target loss.

Model	Target loss	Token Speed-up	Wall-clock Time Speed-up
GPT-124M	3.3159	$\uparrow$ 15%	$\uparrow$ 8.49%
GPT-350M	3.0018	$\uparrow$ 14.3%	$\uparrow$ 7.4%

5.3 Comparison with Muon Variants

In this section, we compare QSD with the other Muon variants, NorMuon (Li et al., 2025) and the Newton-Muon (Du & Su, 2026). As shown in Table 6, QSD consistently outperforms both NorMuon and Newton-Muon across all three runs.

Table 6: Validation performance of GPT-124M on different methods on 4 RTX 5000 Ada GPUs.

Method	Run 1	Run 2	Run 3	Avg. loss
NorMuon	3.3264	3.3280	3.3265	3.3270
Newton Muon	3.3233	3.3237	3.3241	3.3237
QSD	3.3151	3.3161	3.3161	**3.3158**

6 Conclusion

We studied the optimization of quadratic model under the spectral-norm constraint, in order to devise better spectral allocation (and associated spectral subspace update) scheme than Muon for LLM pretraining, thus proposed the QSD algorithm. Analysis shows that even in the isotropic curvature case, the spectral allocation should not be uniform, rather it need to be proportional to the *gradient-to-curvature ratio*. Furthermore, the singular directions of the optimal update should also be modified if the spectral directions of gradient and curvature is not aligned. QSD incurs only mild complexity increase because it uses inversion-free Kronecker-factored curvature and online Frank–Wolfe updates. The resulting QSD method admits simple optimization guarantees and consistently improves validation loss over Muon and recent Muon variants in GPT pre-training. At matched validation loss, QSD achieves up to an 8.49% wall-clock speedup at matched validation loss in our experiments.

AI USE STATEMENT

Generative AI tools, including ChatGPT and Claude, were used during this work to assist with the refinement of theoretical arguments and discussing experimental methodology. They were also used to assist with code debugging and to improve the presentation of the manuscript. All mathematical claims, experimental designs, and conclusions were reviewed and validated by the authors. AI-assisted derivations and code were independently checked against the underlying mathematics and experimental implementations. Generative AI was not used to generate or alter experimental results. The authors take full responsibility for the final content of the paper.

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. In *International Conference on Learning Representations*, volume 2026, pp. 138323–138360, 2026.

Jeremy Bernstein and Laker Newhouse. Old optimizer, new norm: An anthology. *arXiv preprint arXiv:2409.20325*, 2024.

Michael Crawshaw, Chirag Modi, Mingrui Liu, and Robert M Gower. An exploration of non-euclidean gradient descent: Muon and its many variants. *arXiv preprint arXiv:2510.09827*, 2025.

Yihe Dong and Will Sawin. Muon^p: Muon with fractional spectral powers. *arXiv preprint arXiv:2606.13867*, 2026.

Zhehang Du and Weijie Su. The newton-muon optimizer. *arXiv preprint arXiv:2604.01472*, 2026.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly*, 3(1-2):95–110, 1956.

Thomas George, César Laurent, Xavier Bouthillier, Nicolas Ballas, and Pascal Vincent. Fast approximate natural gradient descent in a kronecker factored eigenbasis. *Advances in neural information processing systems*, 31, 2018.

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

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International conference on machine learning*, pp. 427–435. PMLR, 2013.

Keller Jordan, Jeremy Bernstein, Brendan Rappazzo, @fernbear.bsky.social, Boza Vlado, You Jiacheng, Franz Cesista, Braden Koszarsky, and @Grad62304977. modded-nanogpt: Speedrunning the nanogpt baseline, 2024a. URL `https://github.com/KellerJordan/modded-nanogpt`.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. *URL https://kellerjordan. github. io/posts/muon*, 6(3):4, 2024b.

Frederik Kunstner, Philipp Hennig, and Lukas Balles. Limitations of the empirical fisher approximation for natural gradient descent. *Advances in neural information processing systems*, 32, 2019.

Zichong Li, Liming Liu, Chen Liang, Weizhu Chen, and Tuo Zhao. Normuon: Making muon more efficient and scalable. *arXiv preprint arXiv:2510.05491*, 2025.

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.

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

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

Depen Morwani, Itai Shapira, Nikhil Vyas, Eran Malach, Sham Kakade, and Lucas Janson. A new perspective on shampoo's preconditioner. In *International Conference on Learning Representations*, volume 2025, pp. 5802–5822, 2025.

Tianyu Pang, Yujie Fang, Zihang Liu, Shenyang Deng, Lei Hsiung, Shuhua Yu, and Yaoqing Yang. Htmuon: Improving muon via heavy-tailed spectral correction. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 36504–36535, 2026.

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

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained lmos. *arXiv preprint arXiv:2502.07529*, 2025.

Artem Riabinin, Egor Shulgin, Kaja Gruntkowska, and Peter Richtárik. Gluon: Making muon & scion great again!(bridging theory and practice of lmo-based optimizers for llms). *arXiv preprint arXiv:2505.13416*, 2025.

Nicol N Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural computation*, 14(7):1723–1738, 2002.

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

Weijie Su. Isotropic curvature model for understanding deep learning optimization: Is gradient orthogonalization optimal? *arXiv preprint arXiv:2511.00674*, 2025.

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

Shuche Wang, Fengzhuo Zhang, Jiaxiang Li, Dirk Bergemann, and Zhuoran Yang. Why muon outperforms adam: A curvature perspective. *arXiv preprint arXiv:2606.04662*, 2026.

Kaiyue Wen, David Hall, Tengyu Ma, and Percy Liang. Fantastic pretraining optimizers and where to find them. In *International Conference on Learning Representations*, volume 2026, pp. 144731–144838, 2026.

Fangzhou Wu, Rikhav Shah, Sandeep Silwal, and Qiuyi Zhang. Dynmuon: A dynamic spectral shaping view of muon. *arXiv preprint arXiv:2605.17109*, 2026a.

Xiaodong Wu, Wenyi Yu, Chao Zhang, and Philip Woodland. Spectral allocation: Why muon outperforms adam, and how to improve muon. *arXiv preprint arXiv:2608.25990*, 2026b.

Chenrui Xu, Wenjing Yan, and Ying-Jun Angela Zhang. Fismo: Fisher-structured momentum-orthogonalized optimizer. *arXiv preprint arXiv:2601.21750*, 2026.

Ruijie Zhang, Yequan Zhao, Ziyue Liu, Zhengyang Wang, and Zheng Zhang. Muon+: Towards better muon via one additional normalization step. *arXiv e-prints*, pp. arXiv–2602, 2026a.

Yechen Zhang, Shuhao Xing, Junhao Huang, Kai Lv, Yunhua Zhou, Xipeng Qiu, Qipeng Guo, and Kai Chen. Mousse: Rectifying the geometry of muon with curvature-aware preconditioning. *arXiv preprint arXiv:2603.09697*, 2026b.

A ORGANIZATION OF THE APPENDIX

The appendix is organized as follows:

- Sec. B: theoretical guarantees for the QSD inner solver, including the Frank–Wolfe optimality certificate and the $O(1/K)$ convergence rate.
- Sec. C: detailed proofs of the theoretical results on curvature-aware spectral allocation, singular-direction adaptation, the Frank–Wolfe optimality certificate, and inner-loop convergence.
- Sec. D: convergence of the QSD iterates under positive damping.
- Sec. E: monotonic surrogate descent under exact line search and the relation between QSD and the Muon update.
- Sec. F: analysis of the curvature inflation factor and its effect on the spectral allocation.
- Sec. G: mechanism ablations and sensitivity analyses for the QSD-specific hyperparameters.
- Sec. H: implementation details of K-FAC factor estimation.
- Sec. I: details of the GGN-based directional curvature calibration and its finite-difference implementation.
- Sec. J: detailed training settings and hyperparameters used in the GPT pre-training experiments.
- Sec. K: limitations of QSD, its theoretical guarantees, and the current experimental evaluation.

B THEORETICAL GUARANTEES

We analyze the inner Frank–Wolfe solver used by QSD. The Frank–Wolfe gap provides a computable certificate of surrogate suboptimality, while the iterates converge at an $O(1/K)$ rate. We also relate the resulting update to Muon under the same quadratic surrogate. All proofs are deferred to Appendix C.

B.1 OPTIMALITY CERTIFICATE

Consider the convex quadratic surrogate

$$\widehat{q}_\eta(D) = \langle G, D\rangle + \frac{\eta}{2}\langle D, \widehat{\mathcal{C}}[D]\rangle, \qquad \|D\|_2 \le \rho, \tag{22}$$

where $\widehat{\mathcal{C}}$ is positive semidefinite and self-adjoint. Let $\widehat{D}^\star \in \arg\min_{\|D\|_2\le\rho} \widehat{q}_\eta(D)$ denote an optimal solution. At Frank–Wolfe iteration s, define

$$R_s = \nabla_D \widehat{q}_\eta(D_s) = G + \eta\widehat{\mathcal{C}}[D_s]. \tag{23}$$

The linear minimization oracle is

$$S_s \in \arg\min_{\|S\|_2\le\rho} \langle R_s, S\rangle = -\rho\ \mathrm{msgn}(R_s), \tag{24}$$

and the corresponding Frank–Wolfe gap is

$$\mathcal{G}_s = \langle R_s, D_s - S_s\rangle = \langle R_s, D_s\rangle + \rho\|R_s\|_*, \tag{25}$$

where the second equality follows from spectral–nuclear norm duality.

Proposition B.1 (Frank–Wolfe optimality certificate)**.** *For every feasible D_s,*

$$0 \le \widehat{q}_\eta(D_s) - \widehat{q}_\eta(\widehat{D}^\star) \le \mathcal{G}_s. \tag{26}$$

In particular, $\mathcal{G}_s = 0$ if and only if D_s is optimal.

The proof is given in Appendix C.3. Intuitively, $\mathcal{G}_s$ measures how much first-order improvement is still available within the spectral-norm ball. A small gap therefore indicates that no feasible direction can substantially improve the current surrogate value.

Exact line search monotonically decreases the quadratic surrogate, with strict decrease whenever the Frank–Wolfe gap is nonzero. Moreover, with zero initialization, the first Frank–Wolfe atom is exactly the Muon direction. Since the line search optimizes over the segment between zero and this atom, $\widehat{q}_\eta(D_1) \le \widehat{q}_\eta(D_{\mathrm{M}})$, and subsequent Frank–Wolfe steps preserve this inequality by monotonic descent. Formal statements and proofs are provided in Appendix E and Appendix E.1.

B.2 CONVERGENCE OF THE QSD INNER SOLVER

We now specialize to the practical QSD surrogate at a fixed training iteration t:

$$\widehat{q}_t(D) = \langle M_t, D\rangle + \frac{\eta_t\tau}{2}\left[\alpha_t\,\mathrm{tr}\left(D^\top B_t D A_t\right) + d\|D\|_F^2\right], \qquad \|D\|_2 \le \rho. \tag{27}$$

Assume $A_t \succeq 0, B_t \succeq 0, \alpha_t > 0, d \ge 0$, and let $D_t^\star \in \arg\min_{\|D\|_2\le\rho} \widehat{q}_t(D)$ denote an optimal solution.

Theorem B.2 (Inner-loop convergence of QSD)**.** *Let $D_{t,K}$ denote the iterate after $K \ge 1$ Frank–Wolfe refinements with exact line search, starting from any feasible initialization $D_{t,0}$. Then, with $r = \min\{m, n\}$,*

$$\widehat{q}_t(D_{t,K}) - \widehat{q}_t(D_t^\star) \le \frac{8\eta_t\tau r\rho^2\left(\alpha_t\|A_t\|_2\|B_t\|_2 + d\right)}{K+2}. \tag{28}$$

Hence, the inner solver converges to the optimal surrogate value at rate $O(K^{-1})$.

The proof is given in Appendix C.4. This matches the classical $O(1/K)$ convergence rate of Frank–Wolfe for smooth convex objectives (Jaggi, 2013), with the constant here specialized to the spectral-norm feasible set and the K-FAC quadratic curvature.

Because Theorem B.2 holds for any feasible initialization, it also applies directly to the practical warm start $D_{t,0} = D_{t-1,K}$. When $d > 0$, the surrogate is additionally strongly convex, which implies convergence of the iterate itself to the unique inner optimum. The corresponding bound is given in Appendix D.

C Proofs of Theoretical Results

This appendix provides proofs for the results stated in Sections 4 and Appendix B.

C.1 Proof of Proposition 4.1

Let

$$\mathcal{S} = \operatorname{span}\{E_1, \dots, E_p\}.$$

Because every E_i is an eigenmode of the self-adjoint operator $\mathcal{C}$, the subspace $\mathcal{S}$ is invariant under $\mathcal{C}$. Its orthogonal complement $\mathcal{S}^\perp$ is also invariant.

For any feasible D, write

$$D = D_\parallel + D_\perp, \qquad D_\parallel \in \mathcal{S}, \qquad D_\perp \in \mathcal{S}^\perp. \tag{29}$$

Since $G \in \mathcal{S}$,

$$\langle G, D_\perp \rangle = 0.$$

Moreover, self-adjointness and invariance imply

$$\left\langle D_\parallel, \mathcal{C}[D_\perp] \right\rangle = 0.$$

Therefore,

$$\begin{aligned} q_\eta(D) &= \langle G, D_\parallel \rangle + \frac{\eta}{2} \left\langle D_\parallel, \mathcal{C}[D_\parallel] \right\rangle \\ &\quad + \frac{\eta}{2} \left\langle D_\perp, \mathcal{C}[D_\perp] \right\rangle \\ &\geq q_\eta(D_\parallel), \end{aligned} \tag{30}$$

where the inequality follows from $\mathcal{C} \succeq 0$.

We next verify that $D_\parallel$ remains feasible. Write

$$D_\parallel = \sum_{i=1}^{p} d_i E_i, \qquad d_i = \langle E_i, D \rangle = u_i^\top D v_i.$$

For every i,

$$|d_i| = |u_i^\top D v_i| \leq \|D\|_2.$$

Since the matrices $E_i = u_i v_i^\top$ share orthonormal left and right singular vectors,

$$\|D_\parallel\|_2 = \max_i |d_i| \leq \|D\|_2 \leq \rho.$$

Thus $D_\parallel$ is feasible, and there exists an optimal solution entirely in $\mathcal{S}$.

We may therefore write

$$D = -\sum_{i=1}^{p} s_i E_i = -U \operatorname{diag}(s_1, \dots, s_p) V^\top. \tag{31}$$

Its spectral norm is

$$\|D\|_2 = \max_i |s_i|,$$

so feasibility is equivalent to

$$|s_i| \leq \rho \qquad \text{for all } i.$$

Using

$$\mathcal{C}[E_i] = \kappa_i E_i$$

and the orthonormality of the E_i's, the objective becomes

$$q_\eta(D) = \sum_{i=1}^{p} \left(-g_i s_i + \frac{\eta \kappa_i}{2} s_i^2 \right). \tag{32}$$

Because $g_i \geq 0$, an optimizer may be chosen with $s_i \geq 0$. The problem therefore decouples into

$$\min_{0 \leq s_i \leq \rho} \left\{ -g_i s_i + \frac{\eta \kappa_i}{2} s_i^2 \right\}. \tag{33}$$

If $\kappa_i > 0$, the unconstrained minimizer satisfies

$$-g_i + \eta \kappa_i s_i = 0,$$

and therefore

$$s_i^{\mathrm{unc}} = \frac{g_i}{\eta \kappa_i}.$$

Clipping to the feasible interval gives

$$s_i^\star = \min \left\{ \rho, \frac{g_i}{\eta \kappa_i} \right\}.$$

If $\kappa_i = 0$, the mode-wise objective becomes $-g_i s_i$, which is minimized at $s_i^\star = \rho$ whenever $g_i > 0$. If $g_i = 0$, every $s_i \in [0, \rho]$ is optimal, so $s_i^\star = \rho$ may be selected without loss of generality. This proves Proposition 4.1.

C.2 Proof of Proposition 4.2

Proof. Consider

$$\min_{\|D\|_2 \leq \rho} \left\{ \langle G, D \rangle + \frac{\eta}{2} \operatorname{tr}(D^\top D A) \right\}, \tag{34}$$

where

$$G = g\, uv^\top, \qquad A \succ 0, \qquad \|u\|_2 = \|v\|_2 = 1.$$

For any feasible D, decompose

$$D = ux^\top + D_\perp, \qquad u^\top D_\perp = 0. \tag{35}$$

Since $G = g\, uv^\top$,

$$\langle G, D_\perp \rangle = 0. \tag{36}$$

Moreover,

$$\begin{aligned} \operatorname{tr}(D^\top D A) &= x^\top A x + \operatorname{tr}(D_\perp^\top D_\perp A) \\ &\geq x^\top A x, \end{aligned} \tag{37}$$

because $A \succ 0$. Also,

$$\|ux^\top\|_2 = \|x\|_2 \leq \|D\|_2. \tag{38}$$

Therefore, there exists an optimal solution of the form

$$D^\star = ux^{\star\top}, \tag{39}$$

where $x^\star$ solves

$$\min_{\|x\|_2 \leq \rho} \left\{ g\, v^\top x + \frac{\eta}{2} x^\top A x \right\}. \tag{40}$$

The unconstrained minimizer is

$$x_{\mathrm{unc}} = -\frac{g}{\eta} A^{-1} v. \tag{41}$$

By assumption,

$$\|x_{\mathrm{unc}}\|_2 > \rho, \tag{42}$$

so the Euclidean-norm constraint in Eq. 40 is active.

The KKT conditions are therefore

$$gv + \eta A x^\star + \lambda x^\star = 0, \qquad \lambda > 0, \qquad \|x^\star\|_2 = \rho. \tag{43}$$

Solving the stationarity condition gives

$$x^\star = -g(\eta A + \lambda I)^{-1} v. \tag{44}$$

Hence,

$$D^\star = -g\, u \left[(\eta A + \lambda I)^{-1} v\right]^\top, \tag{45}$$

with $\lambda > 0$ chosen so that $\|D^\star\|_2 = \|x^\star\|_2 = \rho$.

It remains to compare the right singular directions. Since G is rank one, its right singular direction is v. The right singular direction of $D^\star$ is proportional to $(\eta A + \lambda I)^{-1} v$.

If these two directions were parallel, then for some scalar $c \neq 0$,

$$(\eta A + \lambda I)^{-1} v = cv. \tag{46}$$

Multiplying by $\eta A + \lambda I$ gives

$$v = c(\eta A + \lambda I)v, \tag{47}$$

and therefore

$$Av = \frac{1/c - \lambda}{\eta} v. \tag{48}$$

Thus, v would be an eigenvector of A.

Consequently, if v is not an eigenvector of A,

$$(\eta A + \lambda I)^{-1} v \not\parallel v, \tag{49}$$

and the right singular direction of the constrained optimum differs from that of the gradient. Since $\lambda > 0$ and $\|D^\star\|_2 = \rho$, this direction change occurs while the spectral-norm constraint is active. □

C.3 Proof of Proposition B.1

Proof. Since $\widehat{q}_\eta$ is convex, for any feasible D,

$$\widehat{q}_\eta(D) \geq \widehat{q}_\eta(D_s) + \langle \nabla_D \widehat{q}_\eta(D_s), D - D_s \rangle. \tag{50}$$

Taking $D = \widehat{D}^\star$ and using $R_s = \nabla_D \widehat{q}_\eta(D_s)$ gives

$$\widehat{q}_\eta(D_s) - \widehat{q}_\eta(\widehat{D}^\star) \leq \langle R_s, D_s - \widehat{D}^\star \rangle. \tag{51}$$

By definition, S_s minimizes the linearized objective over the same feasible set, and therefore

$$\langle R_s, S_s \rangle \leq \langle R_s, \widehat{D}^\star \rangle. \tag{52}$$

Hence,

$$\begin{aligned} \widehat{q}_\eta(D_s) - \widehat{q}_\eta(\widehat{D}^\star) &\leq \langle R_s, D_s - \widehat{D}^\star \rangle && (53)\\ &\leq \langle R_s, D_s - S_s \rangle && (54)\\ &= \mathcal{G}_s. && (55) \end{aligned}$$

The lower bound follows from the optimality of $\widehat{D}^\star$.

It remains to characterize the zero-gap case. Since $\mathcal{G}_s \geq 0$, if $\mathcal{G}_s = 0$, the bound above implies

$$\widehat{q}_\eta(D_s) = \widehat{q}_\eta(\widehat{D}^\star), \tag{56}$$

so D_s is optimal. Conversely, if D_s is optimal, the first-order optimality condition for convex optimization over the feasible set is

$$\langle R_s, D - D_s \rangle \geq 0 \qquad \text{for all feasible } D. \tag{57}$$

In particular, taking $D = S_s$ gives

$$\langle R_s, D_s - S_s \rangle \leq 0. \tag{58}$$

Since the Frank–Wolfe gap is always nonnegative, this implies $\mathcal{G}_s = 0$. □

C.4 Proof of Theorem B.2

Proof. Fix a training iteration t. The practical QSD surrogate is

$$\widehat{q}_t(D) = \langle M_t, D\rangle + \frac{\eta_t \tau}{2}\left[\alpha_t \operatorname{tr}\left(D^\top B_t D A_t\right) + d\|D\|_F^2\right]. \tag{59}$$

Its gradient is

$$\nabla_D \widehat{q}_t(D) = M_t + \eta_t \tau\left(\alpha_t B_t D A_t + dD\right). \tag{60}$$

Define

$$L_t = \eta_t \tau\left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right). \tag{61}$$

For any matrices D and D',

$$\begin{aligned}
&\|\nabla_D \widehat{q}_t(D) - \nabla_D \widehat{q}_t(D')\|_F \\
&\quad = \eta_t \tau \,\|\alpha_t B_t (D - D') A_t + d(D - D')\|_F \\
&\quad \le \eta_t \tau\left(\alpha_t \|B_t\|_2 \|A_t\|_2 + d\right)\|D - D'\|_F \\
&\quad = L_t \|D - D'\|_F.
\end{aligned} \tag{62}$$

Thus, $\widehat{q}_t$ has an L_t-Lipschitz gradient with respect to the Frobenius norm.

Let

$$\mathcal{D} = \{D : \|D\|_2 \le \rho\}. \tag{63}$$

For any $D \in \mathcal{D}$, with $r = \min\{m, n\}$,

$$\|D\|_F \le \sqrt{r}\,\|D\|_2 \le \sqrt{r}\,\rho. \tag{64}$$

Hence, for any $D, D' \in \mathcal{D}$,

$$\|D - D'\|_F \le 2\sqrt{r}\,\rho. \tag{65}$$

The Frank–Wolfe curvature constant of $\widehat{q}_t$ over $\mathcal{D}$ therefore satisfies

$$C_t \le L_t \operatorname{diam}_F(\mathcal{D})^2. \tag{66}$$

Using Eq. 65,

$$C_t \le 4 L_t r \rho^2 \tag{67}$$

$$= 4\eta_t \tau r \rho^2\left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right). \tag{68}$$

Let

$$h_s = \widehat{q}_t(D_{t,s}) - \widehat{q}_t(D_t^\star). \tag{69}$$

The standard Frank–Wolfe descent inequality gives, for any $\gamma \in [0, 1]$,

$$h_{s+1} \le (1 - \gamma) h_s + \frac{\gamma^2}{2} C_t. \tag{70}$$

Exact line search performs at least as well as any prescribed γ.

At the first iteration, choosing $\gamma = 1$ gives

$$h_1 \le \frac{C_t}{2}. \tag{71}$$

For subsequent iterations, the standard Frank–Wolfe (Jaggi, 2013) argument with $\gamma_s = 2/(s + 2)$ yields

$$h_K \le \frac{2C_t}{K + 2}, \qquad K \ge 1. \tag{72}$$

Substituting Eq. 68 gives

$$\boxed{\widehat{q}_t(D_{t,K}) - \widehat{q}_t(D_t^\star) \le \frac{8\eta_t \tau r \rho^2\left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right)}{K + 2}.} \tag{73}$$

This proves the result. □

D ITERATE CONVERGENCE UNDER DAMPING

For any perturbation H, the quadratic form induced by the Hessian of $\widehat{q}_t$ is

$$\left\langle H, \nabla^2 \widehat{q}_t[H] \right\rangle = \eta_t \tau \left[\alpha_t \operatorname{tr}\left(H^\top B_t H A_t\right) + d\|H\|_F^2 \right]. \tag{74}$$

Since $A_t \succeq 0$, $B_t \succeq 0$, and $\alpha_t > 0$,

$$\alpha_t \operatorname{tr}\left(H^\top B_t H A_t\right) \geq 0. \tag{75}$$

Therefore, when $d > 0$,

$$\left\langle H, \nabla^2 \widehat{q}_t[H] \right\rangle \geq \eta_t \tau d \|H\|_F^2. \tag{76}$$

Hence, $\widehat{q}_t$ is $\eta_t \tau d$-strongly convex with respect to the Frobenius norm, and its constrained minimizer $D_t^\star$ is unique.

Because $D_t^\star$ minimizes $\widehat{q}_t$ over the convex feasible set, strong convexity implies

$$\widehat{q}_t(D) - \widehat{q}_t(D_t^\star) \geq \frac{\eta_t \tau d}{2} \|D - D_t^\star\|_F^2. \tag{77}$$

In particular,

$$\|D_{t,K} - D_t^\star\|_F^2 \leq \frac{2\left[\widehat{q}_t(D_{t,K}) - \widehat{q}_t(D_t^\star)\right]}{\eta_t \tau d}. \tag{78}$$

Applying Theorem B.2 gives

$$\|D_{t,K} - D_t^\star\|_F^2 \leq \frac{2}{\eta_t \tau d} \frac{8\eta_t \tau r \rho^2 \left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right)}{K+2} \tag{79}$$

$$= \frac{16 r \rho^2 \left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right)}{d(K+2)}. \tag{80}$$

Taking square roots yields

$$\boxed{\|D_{t,K} - D_t^\star\|_F \leq 4\rho \sqrt{\frac{r\left(\alpha_t \|A_t\|_2 \|B_t\|_2 + d\right)}{d(K+2)}}.} \tag{81}$$

E MONOTONIC SURROGATE DESCENT

Proposition E.1 (Monotonic surrogate descent)**.** *Suppose*

$$\gamma_s \in \arg\min_{\gamma\in[0,1]} \widehat{q}_\eta\left(D_s + \gamma(S_s - D_s)\right). \tag{82}$$

Then

$$\widehat{q}_\eta(D_{s+1}) \le \widehat{q}_\eta(D_s), \tag{83}$$

with strict inequality whenever $\mathcal{G}_s > 0$.

Because $\gamma = 0$ is feasible, exact line search cannot increase the surrogate. When $\mathcal{G}_s > 0$, the Frank–Wolfe direction is strictly descending.

Proof. Let

$$\Delta_s = S_s - D_s, \tag{84}$$

and define the one-dimensional function

$$\phi_s(\gamma) = \widehat{q}_\eta(D_s + \gamma\Delta_s), \qquad \gamma \in [0,1]. \tag{85}$$

Since $\gamma = 0$ is feasible for the line search and γ_s minimizes ϕ_s over $[0,1]$,

$$\widehat{q}_\eta(D_{s+1}) = \phi_s(\gamma_s) \le \phi_s(0) = \widehat{q}_\eta(D_s). \tag{86}$$

For strict descent, the directional derivative at $\gamma = 0$ is

$$\phi_s'(0) = \langle R_s, S_s - D_s\rangle \tag{87}$$

$$= -\mathcal{G}_s. \tag{88}$$

If $\mathcal{G}_s > 0$, then $\phi_s'(0) < 0$. By differentiability, there exists a sufficiently small $\gamma > 0$ such that

$$\phi_s(\gamma) < \phi_s(0). \tag{89}$$

Exact line search therefore satisfies

$$\widehat{q}_\eta(D_{s+1}) < \widehat{q}_\eta(D_s). \tag{90}$$

□

E.1 RELATION TO THE MUON UPDATE

The zero-initialized solver can be compared directly with the Muon update.

Proposition E.2 (One-step surrogate dominance over Muon)**.** *Suppose that $D_0 = 0$ and exact line search is used. Let*

$$D_{\mathrm{M}} = -\rho\ \mathrm{msgn}(G). \tag{91}$$

Then

$$\widehat{q}_\eta(D_1) \le \widehat{q}_\eta(D_{\mathrm{M}}). \tag{92}$$

Moreover, for every $K \ge 1$,

$$\widehat{q}_\eta(D_K) \le \widehat{q}_\eta(D_{\mathrm{M}}). \tag{93}$$

Since $D_0 = 0$, the first Frank–Wolfe atom is $S_0 = D_{\mathrm{M}}$. Exact line search therefore chooses the best point on the segment

$$\{\gamma D_{\mathrm{M}} : \gamma \in [0,1]\}.$$

which contains the full Muon update at $\gamma = 1$. Subsequent Frank–Wolfe steps preserve the inequality by Proposition E.1.

This comparison concerns the quadratic surrogate rather than the exact training loss, and uses zero initialization only to make the relation to Muon explicit.

Proof. Under the zero initialization $D_0 = 0$,

$$R_0 = \nabla_D \widehat{q}_\eta(0) = G. \tag{94}$$

The first Frank–Wolfe atom is therefore

$$S_0 = -\rho \, \mathrm{msgn}(G) = D_{\mathrm{M}}. \tag{95}$$

The first iterate has the form

$$D_1 = D_0 + \gamma_0(S_0 - D_0) = \gamma_0 D_{\mathrm{M}}, \tag{96}$$

where exact line search chooses

$$\gamma_0 \in \arg\min_{\gamma \in [0,1]} \widehat{q}_\eta(\gamma D_{\mathrm{M}}). \tag{97}$$

Since $\gamma = 1$ is feasible for this one-dimensional optimization,

$$\widehat{q}_\eta(D_1) = \min_{\gamma \in [0,1]} \widehat{q}_\eta(\gamma D_{\mathrm{M}}) \le \widehat{q}_\eta(D_{\mathrm{M}}). \tag{98}$$

This proves the one-step comparison.

For every subsequent Frank–Wolfe refinement, Proposition E.1 gives

$$\widehat{q}_\eta(D_K) \le \widehat{q}_\eta(D_{K-1}) \le \cdots \le \widehat{q}_\eta(D_1). \tag{99}$$

Combining the two inequalities yields

$$\widehat{q}_\eta(D_K) \le \widehat{q}_\eta(D_{\mathrm{M}}), \qquad K \ge 1. \tag{100}$$

□

F Effect of Curvature Inflation

We next examine the curvature inflation used in Section 3.3. Consider the aligned setting, and let $g_i \geq 0$ and $\kappa_i \geq 0$ denote the gradient magnitude and curvature of the i-th mode.

Proposition F.1 (Effect of curvature inflation)**.** *With calibration $\alpha > 0$, inflation $\tau \geq 1$, and damping $d \geq 0$, the optimal amplitude of the i-th mode is*

$$s_i^\star(\tau) = \min\left\{\rho,\ \frac{g_i}{\eta\tau(\alpha\kappa_i + d)}\right\}. \tag{101}$$

For $g_i > 0$, define the effective curvature-to-gradient ratio

$$\chi_i = \frac{\eta\rho(\alpha\kappa_i + d)}{g_i}. \tag{102}$$

The mode remains saturated at the spectral-norm boundary if $\tau\chi_i \leq 1$, and becomes curvature-limited if $\tau\chi_i > 1$. Equivalently, the transition occurs at the critical curvature

$$\kappa_i^{\mathrm{crit}}(\tau) = \frac{1}{\alpha}\left(\frac{g_i}{\eta\tau\rho} - d\right), \tag{103}$$

which decreases monotonically with τ:

$$\frac{\partial\kappa_i^{\mathrm{crit}}}{\partial\tau} = -\frac{g_i}{\alpha\eta\rho\tau^2} < 0. \tag{104}$$

Once a mode is curvature-limited, its amplitude decreases as $1/\tau$.

Intuitively, high-curvature directions are more sensitive to finite updates, since the same update magnitude induces a larger second-order change in the objective. Increasing τ therefore makes these sensitive directions more conservative while preserving less curvature-limited directions.

Proof.

$$G = U\,\mathrm{diag}(g_1, \ldots, g_r)V^\top, \qquad g_i \geq 0, \tag{105}$$

and suppose that the curvature operator shares the same matrix modes,

$$\mathcal{C}[u_i v_i^\top] = \kappa_i u_i v_i^\top, \qquad \kappa_i \geq 0. \tag{106}$$

After calibration, damping, and curvature inflation, the effective curvature operator is

$$\widetilde{\mathcal{C}}[D] = \tau\left(\alpha\mathcal{C}[D] + dD\right). \tag{107}$$

An aligned descent direction can be written as

$$D = -U\,\mathrm{diag}(s_1, \ldots, s_r)V^\top, \qquad 0 \leq s_i \leq \rho. \tag{108}$$

Because the singular vectors are orthonormal, $\|D\|_2 \leq \rho$ is equivalent to $0 \leq s_i \leq \rho$ for every mode.

Substituting Eq. 108 into the quadratic surrogate gives

$$q_\eta(D) = \langle G, D\rangle + \frac{\eta}{2}\langle D, \widetilde{\mathcal{C}}[D]\rangle \tag{109}$$

$$= \sum_{i=1}^{r}\left[-g_i s_i + \frac{\eta\tau}{2}(\alpha\kappa_i + d)s_i^2\right]. \tag{110}$$

Hence the optimization separates across modes. For each i, we solve

$$\min_{0\leq s_i\leq\rho}\left\{-g_i s_i + \frac{\eta\tau}{2}(\alpha\kappa_i + d)s_i^2\right\}. \tag{111}$$

When $\alpha\kappa_i + d > 0$, the unconstrained minimizer satisfies

$$-g_i + \eta\tau(\alpha\kappa_i + d)s_i = 0, \tag{112}$$

and is therefore

$$\bar{s}_i = \frac{g_i}{\eta\tau(\alpha\kappa_i + d)}. \tag{113}$$

Restricting this minimizer to the feasible interval $[0, \rho]$ yields

$$s_i^\star(\tau) = \min\left\{\rho,\ \frac{g_i}{\eta\tau(\alpha\kappa_i + d)}\right\}, \tag{114}$$

which proves Eq. 101. If $\alpha\kappa_i + d = 0$ and $g_i > 0$, the scalar objective is linear and is minimized at $s_i^\star = \rho$, consistent with the limiting form above.

For $g_i > 0$, the mode remains saturated at the boundary precisely when

$$\frac{g_i}{\eta\tau(\alpha\kappa_i + d)} \geq \rho. \tag{115}$$

Rearranging gives

$$\tau\frac{\eta\rho(\alpha\kappa_i + d)}{g_i} \leq 1, \tag{116}$$

or equivalently

$$\tau\chi_i \leq 1, \tag{117}$$

where χ_i is defined in Eq. 102. Hence $s_i^\star = \rho$ throughout this regime, so changes in τ do not affect the amplitude as long as the boundary condition remains satisfied.

The same condition can be expressed as a threshold on curvature:

$$\kappa_i \leq \frac{1}{\alpha}\left(\frac{g_i}{\eta\tau\rho} - d\right) = \kappa_i^{\mathrm{crit}}(\tau). \tag{118}$$

Differentiating with respect to τ gives

$$\frac{\partial\kappa_i^{\mathrm{crit}}}{\partial\tau} = -\frac{g_i}{\alpha\eta\rho\tau^2} < 0. \tag{119}$$

Thus increasing τ monotonically lowers the maximum curvature for which a mode can remain boundary-saturated. If $\kappa_i^{\mathrm{crit}}(\tau) < 0$, no mode with $\kappa_i \geq 0$ and the corresponding g_i can satisfy the boundary condition.

Once the mode lies in the curvature-limited regime,

$$s_i^\star(\tau) = \frac{g_i}{\eta\tau(\alpha\kappa_i + d)} = \frac{1}{\tau}\frac{g_i}{\eta(\alpha\kappa_i + d)}. \tag{120}$$

Therefore,

$$\frac{\partial s_i^\star}{\partial\tau} = -\frac{g_i}{\eta\tau^2(\alpha\kappa_i + d)} < 0 \tag{121}$$

whenever $g_i > 0$, so inflation monotonically attenuates every curvature-limited mode.

Consequently, curvature inflation has two effects: it leaves boundary-saturated modes unchanged, while lowering the transition threshold at which modes become controlled by the quadratic penalty. For fixed g_i, modes with larger κ_i cross this threshold earlier as τ increases. This completes the proof.

G Ablation for the Baseline

G.1 Isolating the Effects of QSD

QSD differs from Muon through two coupled effects: singular-direction adaptation and a nonuniform singular spectrum. To isolate the former, we flatten the spectrum of the full QSD update, $D_{\mathrm{QSD}} = U_Q \Sigma_Q V_Q^\top \rightarrow D_{\mathrm{QSD\text{-}Orth}} = \rho U_Q V_Q^\top$. Thus, Muon vs. QSD-Orth evaluates the QSD-selected directions under a flat spectrum, while QSD-Orth vs. QSD captures the additional effect of spectral shaping given those directions. As shown in Table 7, direction adaptation improves validation loss by 0.0048, while restoring the adaptive spectrum yields a further 0.0105 improvement, for a total gain of 0.0153 over Muon. Since directions and singular values are jointly optimized, these comparisons are controlled rather than strictly additive ablations.

Table 7: Mechanism ablations of QSD.

Method	Muon	QSD-Orth	QSD
Val. Loss ↓	3.3312	3.3264(-0.0048)	**3.3159**(-0.0153)

G.2 Ablation for the baseline hyperparameters

We tune the QSD-specific hyperparameters on GPT-124M and reuse the resulting configuration for larger-scale experiments unless otherwise stated. Tables 8 and 9 show the sensitivity to the number of FW steps and K-FAC damping, respectively. Increasing the number of FW steps improves validation loss up to about three steps, after which the gains become marginal. For K-FAC damping, a moderate damping value of 10^{-6} performs best, while both larger and smaller damping slightly degrade validation loss.

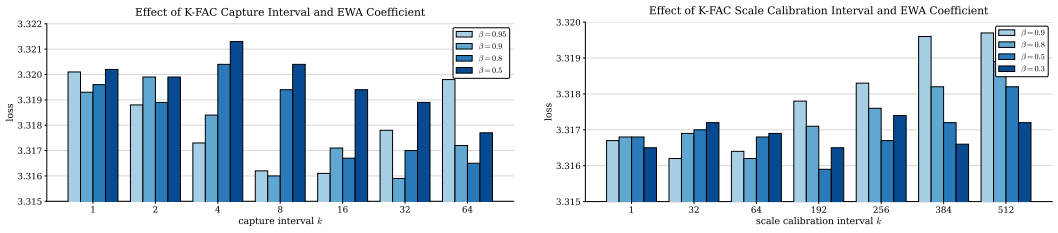
Table 8: Effect of FW Steps (Warm Start).

FW step	1	2	3	4	5
Val. loss	3.3212	3.3199	3.3159	3.3161	3.3155

Table 9: Effect of K-FAC Damping.

d	1e-4	1e-6	1e-8
Val. loss	3.3199	3.3159	3.3182

We further study the effects of the update interval and EMA coefficient for K-FAC factor estimation and curvature-scale calibration. Figure 2a and Figure 2b show a broad region of stable performance, where most variations are small and likely comparable to training noise. However, overly frequent or infrequent updates combined with insufficient smoothing can lead to noticeable degradation.



(a) K-FAC Capture Interval and EMA Coefficient. (b) Scale Calibration Interval and EMA Coefficient.

Figure 2: Sensitivity to K-FAC factor estimation and curvature-scale calibration settings.

Tables 10 and 11 examine the effects of the data sampling ratio used for K-FAC factor estimation and the sample ratio used for scale calibration. For K-FAC factor estimation, using only 1% of the data achieves the best validation loss, with little benefit from larger sampling ratios, and for scale calibration, performance is stable across a wide range of sampling ratios, with 1% already matching or outperforming substantially larger ratios.

Table 10: Effect of K-FAC Sample Ratio.

Ratio	100%	10%	1%	0.5%	0.1%
Val. loss	3.3181	3.3161	3.3159	3.3173	3.3178

Table 11: K-FAC Scale Calibration Sample Ratio.

Ratio	40%	10%	1%	0.5%	0.1%
Val. loss	3.3177	3.3169	3.3159	3.3169	3.3171

Table 12 studies the effect of the curvature inflation factor. Performance improves as τ increases up to 1.5, suggesting that mild curvature inflation benefits training by making curvature-sensitive directions more conservative, while excessive inflation can over-regularize the update.

Table 12: Effect of K-FAC Inflation Factor.

τ	0.75	1.0	1.25	1.5	1.75	2.0
Val. loss	3.3193	3.3173	3.3166	3.3159	3.3168	3.3175

We also provide the effect of the FW step under zero start. As shown in Table 13, for the same number of FW steps, zero start consistently underperforms warm start. With 5 FW steps, zero start still performs worse than warm start with 3 steps.

Table 13: Effect of FW Steps (Zero Start).

FW step	1	2	3	4	5
Val. loss	3.3258	3.3200	3.3183	3.3192	3.3177

H K-FAC Factor Estimation Details

We estimate the K-FAC factors in Eqs. 11–12 from a small subset of token positions in the current training minibatch. The same sampled positions are used for the activation and output-gradient statistics so that each pair $(a_{i,\ell}, \delta_{i,\ell})$ corresponds to the same token.

H.1 Token Sampling

Factor statistics are sampled at the token level. For each microbatch containing N token positions, we uniformly sample without replacement

$$n_{\text{keep}} = \max\{1, \lceil r_F N \rceil\} \tag{122}$$

positions, where r_F is the factor-sampling ratio.

In our default setting $r_F = 0.01$, so each factor estimate uses approximately 1% of the 524288 tokens in an optimizer step (5248 sampled positions). The per-microbatch counts depend on the per-device batch size and the number of devices; under data-parallel training the per-device sufficient statistics $\sum aa^\top$, $\sum \delta\delta^\top$ and the sample counts are summed across devices before normalization, so the pooled estimate is unaffected by the device count.

The factors are averaged over the number of sampled token positions. The same sampling indices are used for $A_{t,\ell}$ and $B_{t,\ell}$ to preserve the correspondence between activations and their output gradients.

H.2 Output-Gradient Rescaling

The output gradients used to construct $B_{t,\ell}$ are obtained from backward hooks. Their raw values contain the scaling introduced by mean-loss reduction, gradient accumulation, AMP loss scaling, and any additional loss scaling used during training. We restore the corresponding per-token scale before computing the K-FAC statistics.

Let $\delta_{i,\ell}^{\text{hook}}$ denote the output gradient captured by the backward hook. The gradient used in the factor estimate is

$$\delta_{i,\ell} = \frac{N\, n_{\text{accum}}}{s_{\text{AMP}}\, s_{\text{custom}}}\, \delta_{i,\ell}^{\text{hook}}, \tag{123}$$

where N is the number of token positions in the original microbatch, n_{accum} is the number of accumulated microbatches, s_{AMP} is the AMP loss-scaling factor, and s_{custom} denotes any additional multiplicative loss scaling. We set $s_{\text{custom}} = 1$ when no additional scaling is used.

Importantly, N in Eq. 123 is the token count before K-FAC subsampling. Subsampling changes only the number of terms used to estimate the factors and does not change the scale of the underlying per-token gradients.

The output-side factor is then computed as

$$\widehat{B}_{t,\ell} = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \delta_{i,\ell} \delta_{i,\ell}^\top, \tag{124}$$

using the same sampled token positions as the corresponding activation-side factor.

H.3 Factor Refresh and EMA

The K-FAC factors are refreshed every $j = 32$ optimizer steps. At a refresh step, the current estimates are incorporated using

$$A_{t,\ell} = \beta_F A_{t-j,\ell} + (1-\beta_F)\widehat{A}_{t,\ell}, \qquad B_{t,\ell} = \beta_F B_{t-j,\ell} + (1-\beta_F)\widehat{B}_{t,\ell}, \tag{125}$$

with $\beta_F = 0.9$.

At the first refresh, the running factors are initialized directly from the first estimates,

$$A_{t,\ell} = \widehat{A}_{t,\ell}, \qquad B_{t,\ell} = \widehat{B}_{t,\ell}. \tag{126}$$

Since the EMA is not initialized from zero, no bias correction is used. Between refresh steps, the most recently stored factors are reused.

H.4 NUMERICAL PRECISION

The sampled activations and output gradients follow the training precision and are stored in bfloat16. The matrix products used to form the K-FAC statistics are therefore computed from bfloat16 operands using the corresponding tensor-core GEMM path. Accumulation across microbatches is performed in FP32, and the resulting K-FAC factors and their EMA states are stored in FP32.

I Directional Curvature Calibration

K-FAC can have a scale mismatch with the corresponding generalized Gauss–Newton (GGN) curvature. We correct this mismatch with a per-layer scalar coefficient by matching the two directional curvatures along an accepted update direction.

Let $\bar{D}_{t,\ell}$ denote the stored update direction used for calibration for layer ℓ, and let θ_t be the parameter point associated with this direction. Both directional curvatures below are evaluated at the same parameter point θ_t.

I.1 Directional Curvature Matching

The K-FAC directional curvature along $\bar{D}_{t,\ell}$ is

$$\begin{aligned} c_{t,\ell}^{\mathrm{KFAC}} &= \left\langle \bar{D}_{t,\ell}, B_{t,\ell}\bar{D}_{t,\ell}A_{t,\ell} \right\rangle \\ &= \mathrm{tr}\left(\bar{D}_{t,\ell}^{\top} B_{t,\ell}\bar{D}_{t,\ell}A_{t,\ell}\right). \end{aligned} \tag{127}$$

Damping is excluded from this calculation. Thus, $\alpha_{t,\ell}$ calibrates only the Kronecker-factored term $B_{t,\ell}DA_{t,\ell}$, while the damping term dD and the inflation factor τ are applied separately.

For the GGN curvature, let $z_i(\theta)$ denote the logits for token i, and let $J_{i,t,\ell}$ be the Jacobian of z_i with respect to the parameters of layer ℓ at θ_t. For softmax cross-entropy,

$$H_{i,t} = \mathrm{diag}(p_{i,t}) - p_{i,t}p_{i,t}^{\top}, \qquad p_{i,t} = \mathrm{softmax}\left(z_i(\theta_t)\right), \tag{128}$$

is the Hessian with respect to the logits. Define

$$v_{i,t,\ell} = J_{i,t,\ell}\,\mathrm{vec}\left(\bar{D}_{t,\ell}\right). \tag{129}$$

The corresponding GGN directional curvature is

$$c_{t,\ell}^{\mathrm{GGN}} = \frac{1}{N_t^{\mathrm{cal}}} \sum_{i \in \mathcal{T}_t^{\mathrm{cal}}} v_{i,t,\ell}^{\top} H_{i,t} v_{i,t,\ell}, \tag{130}$$

where $\mathcal{T}_t^{\mathrm{cal}}$ contains all tokens from the sequences selected for calibration and

$$N_t^{\mathrm{cal}} = |\mathcal{T}_t^{\mathrm{cal}}|. \tag{131}$$

The GGN estimate and the K-FAC statistics use the same per-token averaging convention.

The instantaneous calibration coefficient is

$$\widehat{\alpha}_{t,\ell} = \frac{c_{t,\ell}^{\mathrm{GGN}}}{c_{t,\ell}^{\mathrm{KFAC}}}. \tag{132}$$

Hence, before clipping and temporal smoothing, $\widehat{\alpha}_{t,\ell}$ matches the GGN curvature along $\bar{D}_{t,\ell}$ without changing the Kronecker structure of $A_{t,\ell}$ and $B_{t,\ell}$.

Layers with

$$c_{t,\ell}^{\mathrm{KFAC}} \le \epsilon_{\mathrm{skip}} \tag{133}$$

are skipped for that calibration step and retain their previous calibration coefficient, where

$$\epsilon_{\mathrm{skip}} = 10^{-20}. \tag{134}$$

The instantaneous ratio is clipped before the EMA update:

$$\alpha_{t,\ell} = \beta_\alpha \alpha_{t-k,\ell} + (1-\beta_\alpha)\,\mathrm{clip}\left(\widehat{\alpha}_{t,\ell}, \alpha_{\min}, \alpha_{\max}\right). \tag{135}$$

Between calibration steps, the most recent $\alpha_{t,\ell}$ is reused.

I.2 Finite-Difference GGN Estimate

We do not form the GGN matrix or the Jacobian explicitly. Only the Jacobian–vector product in Eq. 129 is needed, and we approximate it using a forward finite difference.

We first normalize the calibration direction,

$$\widetilde{D}_{t,\ell} = \frac{\bar{D}_{t,\ell}}{\|\bar{D}_{t,\ell}\|_F}. \tag{136}$$

Let $\mathcal{E}_\ell[D]$ denote a perturbation that changes only layer ℓ by D. Then

$$\begin{aligned} v_{i,t,\ell} &= J_{i,t,\ell}\,\mathrm{vec}\left(\bar{D}_{t,\ell}\right) \\ &\approx \frac{\|\bar{D}_{t,\ell}\|_F}{\delta}\left[z_i\left(\theta_t + \delta\,\mathcal{E}_\ell[\widetilde{D}_{t,\ell}]\right) - z_i(\theta_t)\right]. \end{aligned} \tag{137}$$

Substituting Eq. 137 into Eq. 130 gives the GGN directional curvature using only forward evaluations and the closed-form logit-space Hessian in Eq. 128. No explicit Jacobian, GGN matrix, or additional backward pass is required.

The direction and curvature are evaluated at the same parameter point. In the implementation, the parameters are restored to the θ_t associated with $\bar{D}_{t,\ell}$ before computing the unperturbed and perturbed logits. The unperturbed logits $z_i(\theta_t)$ are computed once and shared across the layerwise calibration calculations.

I.3 Implementation Details

We perform curvature calibration every

$$k = 192 \tag{138}$$

training iterations and use

$$\beta_\alpha = 0.5, \qquad \alpha_{\min} = 0.05, \qquad \alpha_{\max} = 100. \tag{139}$$

Calibration data are sampled at the sequence level rather than at the token level. We nominally sample 1% of the sequences in the current minibatch. A minimum calibration block size of 4 sequences is used, and the selected sequences are processed in blocks of 4. With the minibatch size of 512 used in our experiments, the nominal 1% setting results in 8 sequences per calibration step, corresponding to 1.56% of the minibatch. All tokens in each selected sequence are included in the directional-curvature estimate.

For the finite-difference approximation, we use

$$\delta = 0.1 \tag{140}$$

after normalizing the perturbation by $\|\bar{D}_{t,\ell}\|_F$. The finite-difference computation is carried out in FP32 with TF32 disabled. This is important because Eq. 137 subtracts two nearby logit vectors and is sensitive to reduced mantissa precision.

J EXPERIMENTAL DETAILS

Table 14 summarizes the training and QSD-specific hyperparameters used in our main experiments. Unless otherwise stated, QSD-specific hyperparameters tuned on GPT-124M are kept unchanged for GPT-350M.

Table 14: Hyperparameters used in the main experiments.

Hyperparameter	GPT-124M	GPT-350M
Training setup		
Dataset	FineWeb	FineWeb
Sequence length	1024	1024
Batch size (sequences)	512	512
Training tokens	2.5B	7B
Precision	bfloat16	bfloat16
Number of GPUs	4	4
GPU	RTX 5000 Ada	RTX 5000 Ada
Learning-rate schedule	WSD	WSD
Warmup steps / ratio	0	0
Weight decay	0	0
QSD optimizer		
QSD momentum	0.95	0.95
QSD learning rate	0.02	0.015
Spectral radius ρ	1	1
Number of FW steps K	3	3
K-FAC damping d	10^{-6}	10^{-6}
Curvature inflation τ	1.5	1.5
Muon optimizer		
Muon momentum	0.95	0.95
Muon learning rate	0.01	0.01
Spectral radius ρ	1	1
K-FAC factor estimation		
Factor sampling ratio r_F	1%	1%
Factor refresh interval j	32 optimizer steps	128 optimizer steps
Factor EMA β_F	0.9	0.8
Factor storage / EMA precision	FP32	FP32
Curvature-scale calibration		
Calibration interval k	192 optimizer steps	384 optimizer steps
Calibration EMA β_α	0.5	0.5
Calibration clip	$[0.05, 100]$	$[0.05, 100]$
Nominal sequence sampling ratio	1%	1%
Finite-difference step δ	0.1	0.1
Finite-difference precision	FP32 (TF32 disabled)	FP32 (TF32 disabled)
AdamW parameters		
AdamW learning rate	0.0056	0.0032
AdamW β_1	0.9	0.9
AdamW β_2	0.95	0.95

K LIMITATIONS

QSD relies on a local quadratic model and therefore depends on the quality of the curvature approximation. Our implementation uses Kronecker-factored empirical-Fisher statistics with GGN-based scale calibration, which may still miss aspects of the true curvature.

Our theoretical guarantees apply to the quadratic surrogate rather than directly to the exact training loss, since higher-order effects are not modeled.

Finally, our experiments focus on GPT pre-training at the 124M and 350M scales. Evaluating QSD on larger models, other architectures, and post-training settings remains future work.