MUSEC: MOMENTUM SPECTRAL CLIPPING FOR STABLE MUON-TYPE TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Muon has emerged as a highly effective optimizer for large language model training, often achieving superior convergence and performance compared with the widely adopted Adam and AdamW optimizers. Nevertheless, Muon is prone to training instability due to its spectral flattening, manifested by loss spikes and unbounded growth of model weights. Existing approaches primarily rely on weight or attention-logit clipping, which require architecture-specific modifications and do not directly address instability across all model components. We propose MomentUm SpEctral Clipping (Musec), which replaces Muon's spectral flattening with spectral clipping: rather than setting all singular values of the momentum matrix to approximately one, Musec clips singular values that exceed a threshold while preserving the underlying spectral structure of the momentum. Our strategy provides an optimizer-level, architecture-agnostic mechanism for stabilizing Muon training. Theoretically, we establish convergence guarantees for Musec in nonconvex nonsmooth stochastic optimization. Practically, we develop Soft Musec, an efficient implementation that uses a smooth spectral saturation function approximated by coupled Newton–Schulz iterations. Empirically, Soft Musec consistently improves training stability over existing Muon variants across a wide range of learning rates and model sizes, remaining stable in settings where existing Muon variants diverge while matching their performance under well-tuned configurations.

1 INTRODUCTION

Matrix-aware optimizers have recently emerged as strong alternatives to Adam (Kingma & Ba, 2015) and AdamW (Loshchilov & Hutter, 2019) for training large language models. Methods such as Shampoo (Gupta et al., 2018), SOAP (Vyas et al., 2025), Muon (Jordan et al., 2024; Liu et al., 2025), and Scion (Pethick et al., 2025) exploit the two-dimensional structure of weight matrices to achieve faster convergence. Among these, Muon stands out as the simplest and has already been adopted in several large-scale commercial LLM training efforts, including Kimi K2 (Kimi et al., 2025), GLM-5 (Zeng et al., 2026), and DeepSeek-V4 (Xu et al., 2026).

Despite the fast convergence, Muon training is known to suffer from training instability, frequently exhibiting exploding weight norms and divergent attention logits (Kimi et al., 2025). Muon replaces the singular values of its momentum matrix with approximately one, producing an update with a flat singular spectrum. This spectral flattening can inject substantial update magnitude into directions that have relatively small singular values in the original momentum matrix, contributing to unstable weight growth (Kimi et al., 2025, Appendix E). Several approaches have been proposed to address this instability. Logit soft-capping (Gemma et al., 2024) bounds attention scores after computation, but does not prevent the underlying query-key dot products from growing excessively. QK-Norm (Dehghani et al., 2023; Wortsman et al., 2024) normalizes query and key vectors directly, but is incompatible with architectures such as multi-head latent attention (MLA) (Liu et al., 2024), where key matrices are not explicitly materialized during inference. MuonClip (Kimi et al., 2025) takes a different approach by applying weight clipping to the query-key matrices to constrain attention logits. However, it is inherently architecture-specific and only constrains the query-key weight matrices, leaving the value-output and MLP weights entirely unaddressed. These approaches treat the symptoms of Muon's instability rather than its root cause, which lies in Muon's spectral transformation itself.

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.

In this paper, we propose MomentUm SpEctral Clipping (Musec), which replaces Muon's spectral flattening with a spectral clipping operator. While Muon sets all singular values to one, Musec only clips singular values that exceed a threshold. The resulting update preserves the spectral skewness of the momentum matrix, thereby avoiding the amplification of directions associated with relatively small momentum singular values. Unlike existing remedies that target specific architectural components, such as capping attention logits (Gemma et al., 2024) or clipping query-key weights (Kimi et al., 2025), Musec operates directly within the optimizer and is therefore architecture-agnostic, stabilizing all weight matrices uniformly. Moreover, we show that Musec's update admits a principled interpretation as constrained steepest descent: it solves a Frobenius-norm-penalized quadratic subproblem subject to a spectral-norm constraint, complementing the unconstrained steepest descent interpretation of Muon established by Bernstein & Newhouse (2024). We demonstrate the effectiveness of Musec both theoretically and empirically. Despite the growing empirical adoption of Muon-type optimizers, their convergence properties beyond the smooth setting remain poorly understood. In fact, Jiang et al. (2026a) recently showed that Muon does not converge in the general nonconvex nonsmooth setting, highlighting the need for principled modifications to its update rule. Since training objectives in modern deep neural networks are generally nonconvex and may involve nonsmooth components, such as piecewise linear activations and discrete routing mechanisms, we provide convergence guarantees for Musec in the general nonconvex nonsmooth setting. We summarize our main contributions:

- Theoretically, we consider the following stochastic optimization problem:

$$\min_{\boldsymbol{W}\in\mathbb{R}^{m\times n}} f(\boldsymbol{W}) = \mathbb{E}_{\xi\sim\mathcal{P}}[F(\boldsymbol{W};\xi)], \tag{1}$$

  where $\mathcal{P}$ is some unknown distribution, the objective function $f(\boldsymbol{W})$ is ρ-weakly convex and the stochastic component function $F(\boldsymbol{W};\xi)$ is possibly nonconvex and nonsmooth. We show that Musec converges to a (δ,ϵ)-Goldstein stationary point of the objective with $\mathcal{O}(r^{3/2}\delta^{-1}\epsilon^{-3} + \delta^2\rho^3 r^{-3/2} + r^{3/4}\delta^{-1})$ stochastic gradient oracle calls, where $r = \min(m,n)$ is the smaller matrix dimension. As long as the weak convexity parameter satisfies $\rho \le \mathcal{O}(r\delta^{-1}\epsilon^{-1})$, the dominant term of our complexity is $\mathcal{O}(r^{3/2}\delta^{-1}\epsilon^{-3})$. The dependence on the accuracy parameters δ and ϵ matches that of the optimal stochastic first-order method for nonconvex nonsmooth optimization (Cutkosky et al., 2023). While Jiang et al. (2026a) showed that Muon with the exact polar update does not converge in the nonconvex nonsmooth setting, Li & Tsuchiya (2026) proved that finite Newton–Schulz iterations restore convergence through implicit smoothing. We provide a complementary result: Musec converges in the nonconvex nonsmooth setting through explicit spectral clipping, offering both convergence guarantees and improved training stability.

- We further develop an efficient implementation, Soft Musec, which replaces hard spectral clipping with a smooth saturation function and approximates the resulting matrix transformation using coupled Newton-Schulz iterations. Empirically, Soft Musec achieves more stable training than existing Muon variants across multiple datasets, including FineWeb, OpenWebText, and C4, over a substantially wider range of learning rates. Notably, Soft Musec maintains stable convergence in settings where existing Muon variants diverge, while matching their performance under well-tuned configurations.

Concurrent work. Concurrently with our work, Jiang et al. (2026b) proposed spectral clipping as an optimizer-agnostic wrapper, SPECTRA, that can be applied to various base optimizers, including AdamW (Loshchilov & Hutter, 2019) and Signum (Bernstein et al., 2018). Jiang et al. (2026a) established nonconvex nonsmooth convergence guarantees for Pion and Leon, two matrix-aware optimizers derived from adaptive online learning, and showed that Muon with the exact polar update does not converge in this setting. Li & Tsuchiya (2026) subsequently proved that finite Newton–Schulz iterations restore convergence through implicit smoothing of the polar map. We independently identify spectral clipping as a natural replacement for Muon's spectral flattening, motivated by its ability to preserve the spectral structure of momentum matrices while suppressing excessively large singular values. Our work complements these concurrent studies by providing convergence guarantees for spectral clipping-based Muon variants in the nonconvex nonsmooth setting, together with systematic empirical evaluation demonstrating consistent stability improvements across multiple datasets and model sizes.

2 Related Work

Matrix-Aware Optimizers Matrix-aware optimizers exploit the two-dimensional structure of weight matrices to improve upon coordinate-wise methods such as Adam (Kingma & Ba, 2015). Shampoo (Gupta et al., 2018) and SOAP (Vyas et al., 2025) leverage Kronecker-factored preconditioning, while Muon (Jordan et al., 2024) orthogonalizes the momentum matrix via Newton–Schulz iterations, which Bernstein & Newhouse (2024) showed corresponds to steepest descent under the spectral norm. Scion (Pethick et al., 2025) further develops this viewpoint through the Frank–Wolfe framework. Subsequent work has explored variants of Muon, such as NorMuon (Li et al., 2026), which introduces neuron-wise adaptive scaling after orthogonalization. Concurrently, SPECTRA (Jiang et al., 2026b) proposed spectral clipping as an optimizer-agnostic wrapper with convex smooth convergence analysis, and Yukhimchuk et al. (2026) proposed spectral gradient clipping for heavy-tailed noise. On the theoretical side, Shen et al. (2025) and Gluon (Riabinin et al., 2025) established convergence guarantees under smoothness or generalized smoothness, and Yang et al. (2026) analyzed Spectral Descent under convexity and sharpness. Concurrently, Li & Tsuchiya (2026) showed that finite Newton–Schulz iterations restore convergence in the nonsmooth nonconvex setting through implicit smoothing. Our work provides a complementary perspective, establishing convergence guarantees for spectral clipping-based Muon variants in this regime.

Nonconvex Nonsmooth Optimization The foundations of nonsmooth optimization trace back to the seminal work of Clarke (1975) and Goldstein (1977). Non-asymptotic complexity guarantees for general nonconvex nonsmooth settings remained elusive until Zhang et al. (2020) established the first non-asymptotic complexity guarantees for subgradient methods converging to Goldstein stationary points. This breakthrough led to a series of subsequent developments (Davis et al., 2022; Tian et al., 2022; Kornowski & Shamir, 2022; Cutkosky et al., 2023; Jordan et al., 2023). In particular, Cutkosky et al. (2023) introduced the online-to-nonconvex (O2NC) framework, achieving the first optimal convergence rates for stochastic nonconvex nonsmooth optimization. Our analysis considers the class of weakly convex functions (Duchi & Ruan, 2018; Davis & Drusvyatskiy, 2019; Davis & Grimmer, 2019), which encompasses a broad class of objectives arising in neural network training. For this class of problem, Ji & Yuan (2026) showed that the O2NC framework can be derandomized while preserving the optimal $\mathcal{O}(\delta^{-1}\epsilon^{-3})$ complexity. Recently, Jiang et al. (2026a) established nonconvex nonsmooth convergence guarantees for Pion and Leon, two matrix-aware optimizers derived from adaptive online learning over the spectral-norm ball. Notably, they also demonstrated that Muon with the exact polar update does not converge in this setting, while Li & Tsuchiya (2026) proved that finite Newton–Schulz iterations restore convergence through implicit smoothing of the polar map. We provide a complementary result: Musec converges in the nonconvex nonsmooth setting through explicit spectral clipping, offering both convergence guarantees and improved training stability.

3 Preliminaries and Background

In this section, we establish notation, formalize the problem setting and assumptions, and review the Muon optimizer along with its training instability.

3.1 Notations

Throughout this paper, $\|\cdot\|_2$ and $\|\cdot\|_F$ denote the spectral norm and the Frobenius norm of a matrix, respectively, and $\langle\cdot,\cdot\rangle$ denotes the Frobenius inner product. For a set of matrices $\Omega \subseteq \mathbb{R}^{m\times n}$, we denote $\mathrm{dist}(\mathbf{0},\Omega) \coloneqq \inf_{\boldsymbol{M}\in\Omega}\|\boldsymbol{M}\|_F$ and $\mathrm{conv}(\Omega)$ for the convex hull of Ω. For any positive integer N, we abbreviate $[N] = \{1,\dots,N\}$. We use $\mathbb{B}_\delta(\boldsymbol{W}) = \{\boldsymbol{V}\in\mathbb{R}^{m\times n}\colon \|\boldsymbol{V}-\boldsymbol{W}\|_F \le \delta\}$ to denote the closed ball of radius δ centered at $\boldsymbol{W}\in\mathbb{R}^{m\times n}$. For a scalar $w\ge 0$ and a threshold $D>0$, we define the clipping operator as $\mathrm{clip}(w,D) = \min\{w,D\}$. For a diagonal matrix $\boldsymbol{M}\in\mathbb{R}^{r\times r}$, we apply the clipping entrywise:

$$\mathbf{Clip}(\boldsymbol{M},D) = \mathbf{Diag}(\mathrm{clip}(\boldsymbol{M}_{1,1},D),\dots,\mathrm{clip}(\boldsymbol{M}_{r,r},D)),$$

where $\boldsymbol{M}_{i,i}$ is the (i,i)-th entry of the matrix $\boldsymbol{M}$.

3.2 Problem Formulation

We now formalize the problem setting and state the assumptions used throughout this paper. We begin with the definitions of Lipschitz continuity and weak convexity.

Definition 3.1. *We say a function $h\colon \mathbb{R}^{m\times n} \to \mathbb{R}$ is L-Lipschitz continuous if*

$$|h(\boldsymbol{W}_1) - h(\boldsymbol{W}_2)| \le L \left\|\boldsymbol{W}_1 - \boldsymbol{W}_2\right\|_F,$$

for all $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathbb{R}^{m\times n}$.

The Clarke subdifferential (Clarke, 1990) of a Lipschitz function h at $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ is denoted by $\partial h(\boldsymbol{W})$. We next define weakly convex functions.

Definition 3.2. *We say a function $h\colon \mathbb{R}^{m\times n} \to \mathbb{R}$ is ρ-weakly convex for some $\rho > 0$ if the quadratically regularized function $h(\cdot) + (\rho/2)\left\|\cdot\right\|_F^2$ is convex, or equivalently*

$$h(\boldsymbol{W}_1) \ge h(\boldsymbol{W}_2) + \langle \boldsymbol{g}, \boldsymbol{W}_1 - \boldsymbol{W}_2\rangle - \frac{\rho}{2}\left\|\boldsymbol{W}_1 - \boldsymbol{W}_2\right\|_F^2,$$

for all $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathbb{R}^{m\times n}, \boldsymbol{g} \in \partial h(\boldsymbol{W}_2)$.

In the remainder of this paper, we suppose Problem (1) satisfies the following assumptions.

Assumption 3.3. *For any $\xi \sim \mathcal{P}$, the loss function $F(\boldsymbol{W};\xi)$ is L-Lipschitz continuous with respect to its first argument. Furthermore, the objective $f(\boldsymbol{W})$ is ρ-weakly convex.*

Assumption 3.4. *For each $\boldsymbol{W} \in \mathbb{R}^{m\times n}$, we can access to a stochastic oracle $G(\boldsymbol{W};\xi)$ that is an unbiased estimator of a Clarke subgradient such that $\mathbb{E}[G(\boldsymbol{W};\xi)] \in \partial f(\boldsymbol{W})$. Given a positive constant $\sigma > 0$, we further assume bounded variance:*

$$\mathbb{E}[\left\|G(\boldsymbol{W};\xi) - \mathbb{E}[G(\boldsymbol{W};\xi)]\right\|_F^2] \le \sigma^2.$$

We assume that the objective is bounded below, i.e., $f^* = \inf_{\boldsymbol{W}\in\mathbb{R}^{m\times n}} f(\boldsymbol{W}) > -\infty$ and we denote $\Delta_f \coloneqq f(\boldsymbol{W}_0) - f^*$.

For general nonsmooth nonconvex objectives, convergence is standardly measured via Goldstein stationarity (Goldstein, 1977).

Definition 3.5. *The Goldstein δ-subdifferential of a Lipschitz function f at a point $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ is the convex hull of all Clarke subgradients at points in a δ-ball around $\boldsymbol{W}$, i.e.,*

$$\partial_\delta f(\boldsymbol{W}) \coloneqq \operatorname{conv}\left\{\bigcup_{\boldsymbol{V}\in\mathbb{B}_\delta(\boldsymbol{W})} \partial f(\boldsymbol{V})\right\}.$$

A point $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ is called a (δ,ϵ)-stationary point if $\operatorname{dist}(\mathbf{0}, \partial_\delta f(\boldsymbol{W})) \le \epsilon$.

3.3 The Muon Optimizer

Muon (Jordan et al., 2024) is an optimizer designed for matrix-valued parameters. At each iteration, it orthogonalizes the momentum matrix, replacing its nonzero singular values with one and thereby flattening the singular spectrum. The complete procedure is presented in Algorithm 1. Since computing the full SVD is prohibitively expensive in practice, Muon is typically implemented using Newton–Schulz iterations (Higham, 2008) to approximate this orthogonalization (Jordan et al., 2024; Bernstein & Newhouse, 2024).

Training instability of Muon. Let $\boldsymbol{M}_n = \boldsymbol{U}_n\boldsymbol{S}_n\boldsymbol{V}_n^\top$ denote the SVD of the momentum matrix at step n. The Muon update $\boldsymbol{U}_n\boldsymbol{V}_n^\top$ replaces all nonzero singular values of $\boldsymbol{M}_n$ with one, discarding the spectral magnitude information entirely. Consequently, directions associated with small momentum singular values receive updates of the same magnitude as those associated with large singular values, amplifying weakly represented directions and contributing to weight growth and training instability (Kimi et al., 2025, Appendix E). These observations motivate replacing spectral flattening with spectral clipping, which preserves the spectral structure of the momentum while constraining only excessively large singular values.

Algorithm 1: Muon (Jordan et al., 2024)

Input: Initial point $\boldsymbol{W}_0$, momentum parameter $\beta \in [0, 1)$, learning rate η.
for $n = 0, 1, \ldots, N-1$ **do**
  Sample $\xi_n \sim \mathcal{P}$
  $\boldsymbol{G}_n = G(\boldsymbol{W}_n; \xi_n)$
  if $n = 0$ **then**
    $\boldsymbol{M}_0 = \boldsymbol{G}_0$
  else
    $\boldsymbol{M}_n = (1-\beta)\boldsymbol{M}_{n-1} + \beta \boldsymbol{G}_n$
  end
  $(\boldsymbol{U}_n, \boldsymbol{S}_n, \boldsymbol{V}_n) = \mathrm{SVD}(\boldsymbol{M}_n)$
  $\boldsymbol{W}_{n+1} = \boldsymbol{W}_n - \eta \boldsymbol{U}_n \boldsymbol{V}_n^\top$
end

Algorithm 2: Musec: MomentUm SpEctral Clipping

Input: Initial point $\boldsymbol{W}_0$, momentum parameter $\beta \in [0, 1)$, learning rate η, clipping threshold $D > 0$, positive integers K and T
$N = K \times T$
for $n = 0, 1, \ldots, N-1$ **do**
  Sample $\xi_n \sim \mathcal{P}$
  $\boldsymbol{G}_n = G(\boldsymbol{W}_n; \xi_n)$
  if $n = 0$ **then**
    $\widehat{\boldsymbol{M}}_0 = \boldsymbol{G}_0$
  else
    $\widehat{\boldsymbol{M}}_n = (1-\beta)\boldsymbol{M}_{n-1} + \beta \boldsymbol{G}_n$
  end
  $(\boldsymbol{U}_n, \widehat{\boldsymbol{S}}_n, \boldsymbol{V}_n) = \mathrm{SVD}(\widehat{\boldsymbol{M}}_n)$
  $\boldsymbol{S}_n = \mathrm{clip}(\widehat{\boldsymbol{S}}_n, D)$
  $\boldsymbol{M}_n = \boldsymbol{U}_n \boldsymbol{S}_n \boldsymbol{V}_n^\top$
  $\boldsymbol{W}_{n+1} = \boldsymbol{W}_n - \eta \boldsymbol{M}_n$
end
Set $\overline{\boldsymbol{W}}^{(k)} = \frac{1}{T}\sum_{t=0}^{T-1} \boldsymbol{W}_t^{(k)}$ where $\boldsymbol{W}_t^{(k)} = \boldsymbol{W}_{(k-1)T+t}$ for $\forall k \in [K]$.
Return: $\overline{\boldsymbol{W}} \sim \mathrm{Uniform}(\{\overline{\boldsymbol{W}}^{(k)} : k \in [K]\})$.

4 METHODOLOGY

In this section, we introduce MomentUm SpEctral Clipping (Musec) and establish its convergence guarantees for nonconvex nonsmooth optimization.

4.1 THE ALGORITHM

We present the complete procedure of Musec in Algorithm 2. The key difference from Muon lies in Step 13, where we replace the orthogonalized momentum with a spectrally clipped momentum:

$$\boldsymbol{M}_n = \boldsymbol{U}_n \boldsymbol{S}_n \boldsymbol{V}_n^\top, \quad \text{where} \quad \boldsymbol{S}_n = \mathbf{Clip}(\widehat{\boldsymbol{S}}_n, D).$$

Here, $(\boldsymbol{U}_n, \widehat{\boldsymbol{S}}_n, \boldsymbol{V}_n) = \mathrm{SVD}(\widehat{\boldsymbol{M}}_n)$ is the singular value decomposition of the momentum $\widehat{\boldsymbol{M}}_n$, with $\widehat{\boldsymbol{S}}_n$ the diagonal matrix of its singular values, and $D > 0$ is the clipping threshold. Unlike Muon's polar step, which maps all nonzero singular values to one, the clipping operator only truncates singular values exceeding the threshold D while leaving smaller singular values unchanged. Thus, Musec preserves the spectral structure of the momentum while controlling large spectral components.

In contrast to SPECTRA (Jiang et al., 2026b), which applies spectral clipping as a post-processing step to the accumulated momentum while carrying the unclipped momentum state forward, Musec feeds the clipped momentum $\boldsymbol{M}_{n-1}$ back into the exponential moving average (EMA) update at the next step, integrating spectral clipping directly into the momentum recurrence. This ensures that the momentum state itself remains spectrally bounded, allowing singular values to decay in directions where recent gradients are small, rather than being dominated by historical accumulation.

The return procedure in Steps 16–17 partitions the iterates into K consecutive epochs of length T, averages within each epoch, and returns one epoch average uniformly at random. This two-level averaging scheme is standard in the nonconvex nonsmooth optimization literature (Cutkosky et al., 2023; Ji & Yuan, 2026) and is required to establish convergence to Goldstein stationary points. In practice, it is sufficient to return the final iterate $\boldsymbol{W}_n$, as is conventional in deep learning.

4.2 Musec as Constrained Steepest Descent

Bernstein & Newhouse (2024) showed that Muon's polar update solves the unconstrained steepest descent problem under the spectral norm. We establish an analogous characterization for Musec: its spectrally clipped update arises as the solution to a Frobenius-norm-penalized quadratic subproblem subject to a spectral-norm constraint.

Proposition 4.1. *Let $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_L$ be gradient matrices, let $\lambda > 0$ be a sharpness parameter, and let $D > 0$. For each $l = 1, \ldots, L$, consider the problem*

$$\min_{\boldsymbol{\Delta}_l} \left[\langle \boldsymbol{G}_l, \boldsymbol{\Delta}_l \rangle + \frac{\lambda}{2} \|\boldsymbol{\Delta}_l\|_F^2 \right], \quad \text{s.t.} \quad \|\boldsymbol{\Delta}_l\|_2 \le D/\lambda. \tag{2}$$

where $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product and $\boldsymbol{\Delta}_l$ has the same shape as $\boldsymbol{G}_l$. Let $\boldsymbol{G}_l \in \mathbb{R}^{m_l \times n_l}$ have reduced SVD of the form $\boldsymbol{G}_l = \boldsymbol{U}_l \boldsymbol{S}_l \boldsymbol{V}_l^\top$ with $\boldsymbol{S}_l = \mathbf{Diag}(\sigma_1, \ldots, \sigma_{d_l})$ and $d_l = \min(m_l, n_l)$. Then Problem (2) is solved by

$$\boldsymbol{\Delta}_l = -\eta \cdot \boldsymbol{U}_l \hat{\boldsymbol{S}}_l \boldsymbol{V}_l^\top, \quad \text{where} \quad \eta = \frac{1}{\lambda}, \quad \hat{\boldsymbol{S}}_l = \mathbf{Clip}(\boldsymbol{S}_l, D).$$

The spectral-norm constraint $\|\Delta \boldsymbol{W}_l\|_2 \le D/\lambda$ clips singular values that exceed D while preserving smaller ones. Thus, rather than being an ad hoc modification of Muon, Musec admits a principled interpretation as constrained steepest descent under the Frobenius norm with a spectral-norm budget.

4.3 Convergence Analysis

In this subsection, we present the convergence analysis of Musec for solving the nonconvex nonsmooth optimization problem.

We define the filtration $\mathcal{F}_n = \sigma(\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n)$. Under Assumption 3.4, we can infer that $\boldsymbol{H}_n \coloneqq \mathbb{E}[\boldsymbol{G}_n \mid \mathcal{F}_{n-1}] \in \partial f(\boldsymbol{W}_n)$. Analogous to $\boldsymbol{W}_t^{(k)}$, we define variables $\boldsymbol{M}_t^{(k)} = \boldsymbol{M}_{(k-1)T+t}$ and $\boldsymbol{H}_t^{(k)} = \boldsymbol{H}_{(k-1)T+t}$ for any $k \in [K]$ and $t \in \{0\} \cup [T-1]$, and we denote the epoch average $\overline{\boldsymbol{H}}^{(k)} = \sum_{t=1}^{T} \boldsymbol{H}_t^{(k)}/T$. We first establish the following convergence result for a single epoch k.

Lemma 4.2. *Let $\gamma = \beta/\eta$ and $\eta \le 1/\rho$ where $\beta \le 1/8$, then for any $k \in [K]$, the sequence $\{\boldsymbol{W}_t^{(k)}\}_{t=1}^T$ generated by Algorithm 2 satisfies*

$$\begin{aligned}
&\mathbb{E}\left[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)}) \right] \\
&\le -\mathbb{E}\left[\frac{\beta D T}{\gamma} \left\| \overline{\boldsymbol{H}}^{(k)} \right\|_F + \sum_{t=0}^{T-1} \frac{\beta}{8\gamma} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2 \right] + \frac{\beta D \sigma \sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{1}{\gamma} \right) D^2 + \frac{\beta^2 L^2 T}{\gamma} + \frac{r D^2}{\gamma}.
\end{aligned}$$

By combining Lemma 4.2 with a bound relating $\|\overline{\boldsymbol{H}}^{(k)}\|_F$ to the Goldstein δ-subdifferential $\mathrm{dist}(0, \partial_\delta f(\overline{\boldsymbol{W}}_k))$, we obtain the following convergence guarantee for Musec (Algorithm 2) in finding a (δ, ϵ)-Goldstein stationary point of nonconvex nonsmooth problem.

Theorem 4.3. *Let $\gamma = \beta/\eta$ and $\eta \le 1/\rho$ where $\beta \le 1/8$, Then for any $\delta \ge \eta T D\sqrt{r}$, the sequence $\{\overline{\boldsymbol{W}}^{(k)}\}_{k=1}^{K}$ generated by Algorithm 2 satisfies*

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\operatorname{dist}(\mathbf{0}, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)}))\right] \le \frac{\sigma}{\sqrt{T}} + \left(1 + \frac{2r}{\beta T}\right) D + \frac{\beta L^2}{D} + \frac{\gamma \Delta_f}{\beta D T K}.$$

The following corollary, which follows directly from Theorem 4.3, establishes the oracle complexity of Musec for finding a (δ, ϵ)-Goldstein stationary point.

Corollary 4.4. *By choosing the parameters as*

$$T = (\delta N)^{2/3}, \quad K = \delta^{-2/3} N^{1/3}, \quad D = (\delta N)^{-1/3},$$
$$\beta = \frac{r^{1/2}}{(\delta N)^{2/3}}, \quad \gamma = \frac{r}{\delta^{4/3} N^{1/3}}, \quad \eta = \frac{\delta^{2/3}}{\sqrt{r} N^{1/3}},$$

then it takes Algorithm 2 at most

$$N = \mathcal{O}\left(\frac{r^{3/4}}{\delta} + \frac{\delta^2 \rho^3}{r^{3/2}} + \frac{r^{3/2}(\sigma^3 + L^6 + \Delta_f^3)}{\delta \epsilon^3}\right)$$

to obtain a (δ, ϵ)-Goldstein stationary point.

Remark 4.5. When the weak convexity parameter satisfies $\rho \le \mathcal{O}(r\delta^{-1}\epsilon^{-1})$, the dominant term of the complexity reduces to $\mathcal{O}(r^{3/2}\delta^{-1}\epsilon^{-3})$. The dependence on the accuracy parameters δ and ϵ matches that of the optimal stochastic first-order method for nonconvex nonsmooth optimization (Cutkosky et al., 2023). The factor of $r^{3/2}$ arises from the matrix-valued structure of the updates. For comparison, the convergence analysis of Muon under smoothness assumptions by Shen et al. (2025) incurs a factor of r^2 in the stochastic setting, though we note that the two results target different stationarity concepts and operate under different assumptions.

5 A Practical Implementation

Computing the exact spectral clipping operator requires a full SVD decomposition at each iteration (step 11 in Algorithm 2), which is prohibitively expensive for the large weight matrices encountered in LLM training. In this section, we develop an efficient SVD-free approximation of the clipping operation. We term the resulting algorithm Soft Musec.

Specifically, we replace the hard clipping function $\operatorname{clip}(w, D) = \min(w, D)$ with the smooth saturation function

$$h(w, D) = \frac{Dw}{\sqrt{w^2 + D^2}},$$

which satisfies that $h(w, D) \approx w$ when $w \ll D$ and $h(w, D) \approx D$ when $w \gg D$. Applying $h(\cdot, D)$ to each singular value of a matrix $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ while preserving its singular vectors yields the soft spectral clipping operator:

$$H(\boldsymbol{M}, D) = D(\boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I})^{-1/2}\boldsymbol{M}. \tag{3}$$

The key computational challenge in evaluating the operator (3) is the matrix inverse square root $(\boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I})^{-1/2}$. We approximate this using coupled Newton–Schulz iterations (Higham 2008, Chapter 6; Jiang et al. 2026b), which involve only matrix–matrix multiplications and are therefore well-suited to modern GPU hardware. The addition of $D^2\boldsymbol{I}$ shifts all eigenvalues away from zero, ensuring numerical stability and rapid convergence of the iteration. In practice, we find that five Newton–Schulz iterations yield strong training performance in bfloat16 arithmetic. The complete Soft Musec algorithm is provided in Algorithm 4 in Appendix C.

6 Experiments

In this section, we evaluate the effectiveness and stability of Soft Musec on LLM training.

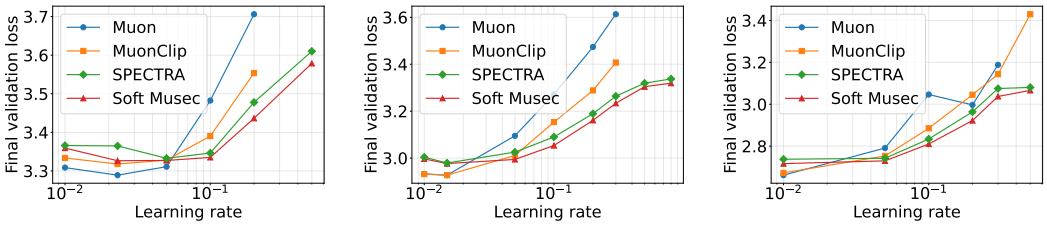
6.1 Experimental Setup

We compare Soft Musec against three baselines: Muon (Jordan et al., 2024), MuonClip (Kimi et al., 2025), and SPECTRA (Jiang et al., 2026b). For SPECTRA, we use stochastic gradient descent with momentum (SGDM) as the base optimizer, which corresponds to Muon without the orthogonalization step. This isolates the effect of spectral clipping from Muon's orthogonalization. We note that Jiang et al. (2026b) applied SPECTRA on AdEMAMix (Pagliardini et al., 2025), AdamW (Loshchilov & Hutter, 2019), and Signum (Bernstein et al., 2018). To our knowledge, our experiments provide the first empirical evaluation of spectral clipping applied to Muon in this setting. For all methods, we tune weight decay over $\{0.096, 0.3, 0.6, 1.2\}$. All methods use five Newton–Schulz iterations. For Soft Musec and SPECTRA, we additionally tune the clipping threshold D over $\{0.05, 0.125, 0.25, 0.5, 0.75, 1.0\}$.

We adopt the modded-nanogpt codebase[1], a decoder-only Transformer with modern architectural modifications, including RMSNorm (Zhang & Sennrich, 2019), Rotary Positional Embeddings (RoPE) (Su et al., 2024), and square ReLU activations (So et al., 2021). We evaluate three model configurations—NanoGPT-Small (491M parameters), NanoGPT-Medium (613M parameters), and NanoGPT-Wide (1.63B parameters)—across three datasets: FineWeb (Penedo et al., 2024), OpenWebText (Gokaslan & Cohen, 2019), and C4 (Raffel et al., 2020). For all experiments, the respective optimizer is applied to matrix parameters in non-embedding layers (attention and MLP weight matrices), while AdamW is used for embedding layers and vector parameters. Full architectural and hyperparameter details are provided in Appendix E.

6.2 Stability Across Learning Rates

We evaluate each method across learning rates ranging from 0.01 to 0.8 for each model configuration on FineWeb. The exact grid for each configuration is provided in Appendix E.3. Results are shown in Figure 1; runs that diverge during training are omitted. For NanoGPT-Small, all methods achieve comparable validation loss at smaller learning rates. As the learning rate increases, Soft Musec maintains stable convergence, while Muon and MuonClip begin to degrade. At the largest learning rate, Muon fails to converge across all model sizes. MuonClip also fails to converge on NanoGPT-Small and NanoGPT-Medium, and converges to substantially worse validation loss on NanoGPT-Wide. In contrast, Soft Musec and SPECTRA remain stable across all configurations. Across all configurations, Soft Musec achieves performance comparable to SPECTRA, with slight improvements in several settings. Consistent results on OpenWebText and C4 are reported in Appendix F.1.



(a) NanoGPT-small (b) NanoGPT-medium (c) NanoGPT-wide

Figure 1: Validation loss versus learning rate on FineWeb across three model configurations.

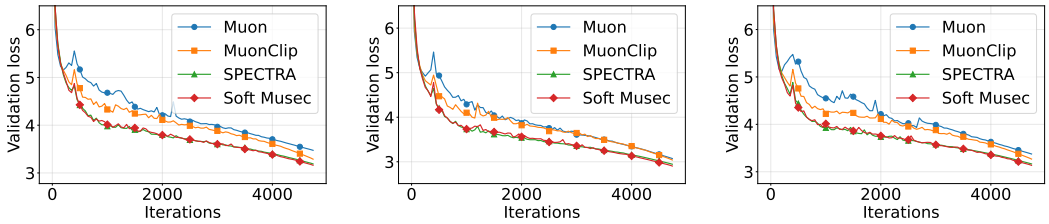
6.3 Training Dynamics

To examine training behavior in greater detail, we train NanoGPT-Medium at a learning rate of 0.2 on all three datasets and plot the validation loss over training steps in Figure 2.

All methods exhibit a transient loss spike near step 400, which coincides with a scheduled transition in the modded-nanogpt training pipeline: the learning rate increases by 52%, the batch size doubles, and the attention window widens simultaneously. Notably, Soft Musec and SPECTRA recover from

[1] https://github.com/kellerjordan/modded-nanogpt

this spike quickly and smoothly, while Muon recovers slowly and converges to a substantially higher final validation loss. MuonClip improves over Muon but displays noticeable oscillations throughout training, suggesting that QK-weight clipping alone does not fully stabilize the optimization trajectory. In contrast, both Soft Musec and SPECTRA converge smoothly and reach comparable final validation loss. Complete training dynamics across all datasets and model configurations are reported in Appendix F.2.

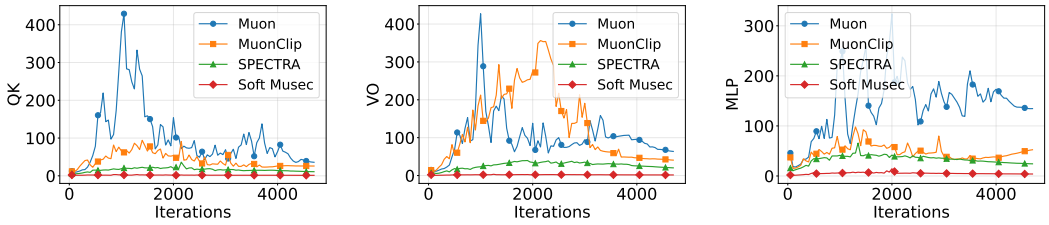


(a) FineWeb (b) OpenWebText (c) C4

Figure 2: Validation loss versus steps of training NanoGPT-Medium on the three datasets.

6.4 Weight Norm Analysis

To understand the source of the stability improvements, we track the spectral norm of weight matrices across three component types: query-key (QK) projections, value-output (VO) projections, and MLP weights, during NanoGPT-Medium training on FineWeb at a learning rate of $\eta = 0.2$. Results are shown in Figure 3. Muon exhibits severe spectral-norm inflation across all three components, with norms reaching 300–400 and displaying large oscillations. This is consistent with the spectral flattening mechanism discussed in Section 3: by assigning comparable update magnitudes to all singular directions, Muon amplifies weakly represented directions, leading to substantial weight growth. The unchecked growth of QK norms is particularly concerning, as it can drive attention logit explosion (Kimi et al., 2025). MuonClip partially mitigates QK norm growth but leaves VO and MLP norms elevated, confirming that it only regulates the query-key matrices. In contrast, both Soft Musec and SPECTRA maintain spectral norms below 20 across all components, demonstrating that optimizer-level spectral clipping provides uniform stabilization across the entire model.



(a) QK projections (b) VO projections (c) MLP weights

Figure 3: Spectral norm of weight matrices over training steps for NanoGPT-Medium on FineWeb.

7 Conclusion

In this paper, we introduced Musec, which replaces Muon's spectral flattening with a spectral clipping operator that preserves the spectral structure of the momentum matrix while constraining excessively large singular values. We established the convergence analysis of Musec in the nonconvex nonsmooth setting, with a rate matching the optimal dependence on the accuracy parameters for stochastic nonconvex nonsmooth optimization. On the practical side, we developed Soft Musec, an efficient SVD-free implementation based on coupled Newton–Schulz iterations, and demonstrated that it substantially improves training stability over existing Muon variants across multiple datasets and model sizes.

AI USE STATEMENT

In this work, we used generative AI tools for polishing the writing / implement methods / design or provide feedback on research methodology or experiments / interpret results / formulate mathematical claims. We have not used generative AI tools for developing theoretical models or conceptual frameworks / providing critical ingredients for proving mathematical claims / support qualitative and thematic data analysis / provide critical ingredients for proving mathematical claims / assist in the writing of proofs / propose or refine hypotheses and generate synthetic data sets / assist with translation / clean and reformat dataset are not applicable to this work. Additionally, we used generative AI tools for suggest experimental parameters / create or edit software code / summarize or analyse existing literature / discover research topics or identify gaps. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness by us, LLM-recommended references are double-checked by us. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We take several steps to ensure reproducibility. All theoretical results, including the convergence analysis of Musec (Theorem 4.3 and Corollary 4.4) and the constrained steepest descent characterization (Proposition 4.1), are stated with explicit assumptions and proved in full in Appendices B and A, respectively. On the experimental side, we provide complete architecture specifications (Table 1), training configurations (Table 2), and per-method hyperparameter choices (Tables 3–5) in Appendix E. The Soft Musec algorithm is described in full in Algorithm 4 (Appendix C), and the core implementation is provided in the supplementary materials.

REFERENCES

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

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signSGD: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pp. 560–569. PMLR, 2018.

Frank H. Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

Frank H. Clarke. *Optimization and Nonsmooth Analysis*, volume 5 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1990.

Ashok Cutkosky, Harsh Mehta, and Francesco Orabona. Optimal stochastic non-smooth non-convex optimization through online-to-non-convex conversion. In *International Conference on Machine Learning*, pp. 6643–6670. PMLR, 2023.

Damek Davis and Dmitriy Drusvyatskiy. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Damek Davis and Benjamin Grimmer. Proximally guided stochastic subgradient method for nonsmooth, nonconvex problems. *SIAM Journal on Optimization*, 29(3):1908–1930, 2019.

Damek Davis, Dmitriy Drusvyatskiy, Yin Tat Lee, Swati Padmanabhan, and Guanghao Ye. A gradient sampling method with complexity guarantees for lipschitz functions in high and low dimensions. *Advances in Neural Information Processing Systems*, 35:6692–6703, 2022.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pp. 7480–7512. PMLR, 2023.

John Duchi and Feng Ruan. Stochastic methods for composite and weakly convex optimization problems. *SIAM Journal on Optimization*, 28(4):3229–3259, 2018.

Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International Conference on Machine Learning*, pp. 10929–10974. PMLR, 2023.

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

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`, 2019.

Allen A Goldstein. Optimization of lipschitz continuous functions. *Mathematical Programming*, 13 (1):14–22, 1977.

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

Nicholas J. Higham. *Functions of Matrices: Theory and Computation*. SIAM, 2008.

Fanfan Ji and Xiaotong Yuan. Derandomized online-to-non-convex conversion for stochastic weakly convex optimization. In *International Conference on Learning Representations*, volume 2026, pp. 125389–125414, 2026.

Ruichen Jiang, Zakaria Mhammedi, Mehryar Mohri, and Aryan Mokhtari. Adaptive matrix online learning through smoothing with guarantees for nonsmooth nonconvex optimization. *arXiv preprint arXiv:2602.08232*, 2026a.

Xiaowen Jiang, Andrei Semenov, and Sebastian U. Stich. Enhancing llm training via spectral clipping. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026b.

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

Michael Jordan, Guy Kornowski, Tianyi Lin, Ohad Shamir, and Manolis Zampetakis. Deterministic nonsmooth nonconvex optimization. In *The Thirty-Sixth Annual Conference on Learning Theory*, pp. 4570–4597. PMLR, 2023.

Team Kimi, Yifan Bai, Yiping Bao, Y Charles, Cheng Chen, Guanduo Chen, Haiting Chen, Huarong Chen, Jiahao Chen, Ningxin Chen, et al. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*, 2025.

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

Guy Kornowski and Ohad Shamir. Oracle complexity in nonsmooth nonconvex optimization. *Journal of Machine Learning Research*, 23(314):1–44, 2022.

Mingyi Li and Taira Tsuchiya. Muon with finite Newton-Schulz: The smoothing benefit in nonsmooth nonconvex optimization. *arXiv preprint arXiv:2608.26288*, 2026.

Zichong Li, Liming Liu, Chen Liang, Weizhu Chen, and Tuo Zhao. NorMuon: Making Muon more efficient and scalable. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 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.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Matteo Pagliardini, Pierre Ablin, and David Grangier. The AdEMAMix optimizer: Better, faster, older. In *International Conference on Learning Representations*, volume 2025, pp. 64715–64757, 2025.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 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. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 49069–49104, 2025.

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

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.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, pp. 606–610. IEEE, 2007.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. FlashAttention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685, 2024.

Wei Shen, Ruichuan Huang, Minhui Huang, Cong Shen, and Jiawei Zhang. On the convergence analysis of muon. *arXiv preprint arXiv:2505.23737*, 2025.

David So, Wojciech Mańke, Hanxiao Liu, Zihang Dai, Noam Shazeer, and Quoc V Le. Searching for efficient transformers for language modeling. *Advances in neural information processing systems*, 34:6010–6022, 2021.

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

Lai Tian, Kaiwen Zhou, and Anthony Man-Cho So. On the finite-time complexity and practical computation of approximate stationarity concepts of Lipschitz functions. In *International Conference on Machine Learning*, pp. 21360–21379. PMLR, 2022.

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.

Mitchell Wortsman, Peter Liu, Lechao Xiao, Katie Everett, Alexander Alemi, Ben Adlam, John D Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, et al. Small-scale proxies for large-scale transformer training instabilities. In *International Conference on Learning Representations*, volume 2024, pp. 49844–49869, 2024.

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

Yixuan Yang, Yuqing He, and Song Li. Convergence of spectral descent for non-smooth optimization. *arXiv preprint arXiv:2605.26977*, 2026.

Alexander Yukhimchuk, Mladen Kolar, Martin Takáč, and Sayantan Choudhury. Gradient clipping beyond vector norms: A spectral approach for matrix-valued parameters. *arXiv preprint arXiv:2605.11838*, 2026.

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

Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

Jingzhao Zhang, Hongzhou Lin, Stefanie Jegelka, Suvrit Sra, and Ali Jadbabaie. Complexity of finding stationary points of nonconvex nonsmooth functions. In *International Conference on Machine Learning*, pp. 11173–11182. PMLR, 2020.

APPENDIX

The appendix is organized as follows. Section A proves that Musec's update solves a Frobenius-norm-penalized quadratic subproblem subject to a spectral-norm constraint. Section B provides the detailed convergence analysis of Musec for nonconvex nonsmooth optimization. Section C presents the complete Soft Musec algorithm with coupled Newton–Schulz iterations. Section D compares the update rules of Musec and SPECTRA. Section E details the architecture configurations, training configurations, and hyperparameter choices for the NanoGPT experiments. Section F provides additional learning rate sweep results, training dynamics, effective rank analysis, computational overhead, and ablation studies.

A PROOF OF PROPOSITION 4.1

In this section, we give a formal proof of Proposition 4.1. We first restate the proposition as follows.

Proposition A.1. *Let $\boldsymbol{G}_1, \dots, \boldsymbol{G}_L$ be gradient matrices, let $\lambda > 0$ be a sharpness parameter, and let $D > 0$. For each $l = 1, \dots, L$, consider the problem*

$$\arg\min_{\boldsymbol{\Delta}_l} \left[\langle \boldsymbol{G}_l, \boldsymbol{\Delta}_l \rangle + \frac{\lambda}{2} \|\boldsymbol{\Delta}_l\|_F^2 \right], \quad \text{s.t.} \quad \|\boldsymbol{\Delta}_l\|_2 \le \frac{D}{\lambda}. \tag{4}$$

where $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product and $\boldsymbol{\Delta}_l$ has the same shape as $\boldsymbol{G}_l$. Let $\boldsymbol{G}_l \in \mathbb{R}^{m_l \times n_l}$ have reduced SVD of the form $\boldsymbol{G}_l = \boldsymbol{U}_l \boldsymbol{S}_l \boldsymbol{V}_l^\top$ with $\boldsymbol{S}_l = \mathbf{Diag}(\sigma_1, \dots, \sigma_{d_l})$ and $d_l = \min(m_l, n_l)$. Then Problem (2) is solved by

$$\boldsymbol{\Delta}_l = -\eta \cdot \boldsymbol{U}_l \hat{\boldsymbol{S}}_l \boldsymbol{V}_l^\top, \quad \text{where} \quad \eta = \frac{1}{\lambda}, \quad \hat{\boldsymbol{S}}_l = \mathbf{Clip}(\boldsymbol{S}_l, D).$$

Proof. The Frobenius norm $\|\boldsymbol{\Delta}_l\|_F$ and the spectral norm $\|\boldsymbol{\Delta}_l\|_2$ depend on $\boldsymbol{\Delta}_l$ only through its singular values. By Von Neumann's trace inequality, the inner product $\langle \boldsymbol{G}_l, \boldsymbol{\Delta}_l \rangle$ is minimized, for any fixed spectrum of $\boldsymbol{\Delta}_l$, by aligning $\boldsymbol{\Delta}_l$'s singular vectors with those of $-\boldsymbol{G}_l$. It therefore suffices to restrict attention to

$$\boldsymbol{\Delta}_l = -\boldsymbol{U}_l \boldsymbol{A}_l \boldsymbol{V}_l^\top, \quad \boldsymbol{A}_l = \mathrm{diag}(a_1, \dots, a_{d_l}), \quad a_1 \ge a_2 \ge \cdots \ge a_{d_l} \ge 0.$$

Under this parametrization,

$$\langle \boldsymbol{G}_l, \boldsymbol{\Delta}_l \rangle = -\sum_{i=1}^{d_l} \sigma_i a_i, \qquad \|\boldsymbol{\Delta}_l\|_F^2 = \sum_{i=1}^{d_l} a_i^2, \qquad \|\boldsymbol{\Delta}_l\|_2 = a_1.$$

Since $a_1 \ge \cdots \ge a_{d_l} \ge 0$, the constraint $a_1 \le D/\lambda$ is equivalent to $0 \le a_i \le D/\lambda$ for every i, and Problem (4) reduces to the separable box-constrained program

$$\min_{a_i \in [0, D/\lambda]} \sum_{i=1}^{d_l} \left[-\sigma_i a_i + \frac{\lambda}{2} a_i^2 \right]. \tag{5}$$

Each scalar subproblem is a quadratic in a_i with an unconstrained minimizer $\sigma_i/\lambda \ge 0$; clipping to the interval $[0, D/\lambda]$ gives

$$a_i^* = \min\left(\frac{\sigma_i}{\lambda}, \frac{D}{\lambda} \right) = \frac{1}{\lambda} \min(\sigma_i, D).$$

The ordering $\sigma_1 \ge \cdots \ge \sigma_{d_l}$ ensures $a_1^* \ge \cdots \ge a_{d_l}^*$, so the monotonicity of the parametrization is automatically satisfied. Since $a_i^* = \min(\sigma_i, D)/\lambda$, we have $\boldsymbol{A}_l^* = \frac{1}{\lambda} \hat{\boldsymbol{S}}_l$ with $\hat{\boldsymbol{S}}_l = \mathbf{Clip}(\boldsymbol{S}_l, D)$, and hence

$$\boldsymbol{\Delta}_l = -\boldsymbol{U}_l \boldsymbol{A}_l^* \boldsymbol{V}_l^\top = -\eta \cdot \boldsymbol{U}_l \hat{\boldsymbol{S}}_l \boldsymbol{V}_l^\top,$$

with $\eta = 1/\lambda$, as claimed. □

B CONVERGENCE ANALYSIS

In this section, we provide the detailed convergence analysis of the Musec algorithm.

B.1 Regret Analysis of Online Gradient Descent

Before presenting the convergence analysis of the Musec algorithm, we first establish a convergence result for online gradient descent on a quadratic objective, which will subsequently serve as the basis for analyzing the clipping operation in Musec.

Consider a quadratic function of the form:

$$f_t(\boldsymbol{X}) \coloneqq -\frac{\beta}{\gamma}\langle \boldsymbol{G}_{t+1}, \boldsymbol{X}\rangle + \frac{\beta}{2\gamma}\|\boldsymbol{X}\|_F^2$$

in the matrix-valued variable $\boldsymbol{X}$. We analyze the following standard online gradient descent (OGD) method with stepsize $\gamma > 0$ over a convex compact set $\mathcal{C}$:

$$\begin{aligned}\boldsymbol{X}_{t+1} \coloneqq& \Pi_{\mathcal{C}}(\boldsymbol{X}_t - \gamma\nabla f_t(\boldsymbol{X}_t))\\ =&\Pi_{\mathcal{C}}(\boldsymbol{X}_t + \beta\boldsymbol{G}_{t+1} - \beta\boldsymbol{X}_t)\\ =&\Pi_{\mathcal{C}}\left((1-\beta)\boldsymbol{X}_t + \beta\boldsymbol{G}_{t+1}\right),\end{aligned} \tag{6}$$

where $\Pi_{\mathcal{C}}(\cdot)$ denotes the Euclidean Projection associated with the convex and compact set $\mathcal{C}$. The OGD step (6) is consistent with the update of the momentum variable $\boldsymbol{M}_n$ in Algorithm 2 when we choose $\mathcal{C} = \{\boldsymbol{X} : \|\boldsymbol{X}\| \le D\}$.

Let $\mathrm{Regret}_T(\overline{\boldsymbol{X}})$ denote the regret of the OGD algorithm with respect to some $\overline{\boldsymbol{X}} \in \mathcal{C}$ after T iterations, defined as:

$$\mathrm{Regret}_T(\overline{\boldsymbol{X}}) \coloneqq \sum_{t=0}^{T-1} f_t(\boldsymbol{X}_t) - \sum_{t=0}^{T-1} f_t(\overline{\boldsymbol{X}}).$$

We present the regret analysis of the OGD algorithm as follows.

Lemma B.1. *Suppose that $\beta \le 1/8$, then the OGD update (6) applied to the sequence of quadratic functions $\{f_t(\boldsymbol{X})\}_{t=0}^{T-1}$ over the convex constraint set $\mathcal{C}$ guarantees that for all $T \ge 1$ and all $\overline{\boldsymbol{X}} \in \mathcal{C}$:*

$$\begin{aligned}&\mathrm{Regret}_T(\overline{\boldsymbol{X}})\\ \le&\sum_{t=0}^{T-1}\left(-\frac{\beta}{8\gamma}\|\boldsymbol{X}_t\|_F^2 + \frac{\beta}{2\gamma}\left\|\overline{\boldsymbol{X}}\right\|_F^2 + \frac{\beta^2}{\gamma}\|\boldsymbol{G}_{t+1}\|_F^2\right) + \frac{\|\boldsymbol{X}_0\|_F^2 + \left\|\overline{\boldsymbol{X}}\right\|_F^2}{\gamma}.\end{aligned}$$

Proof. Fix any $\overline{\boldsymbol{X}} \in \mathcal{C}$. By the non-expansiveness of the projection operator,

$$\begin{aligned}&\left\|\boldsymbol{X}_{t+1} - \overline{\boldsymbol{X}}\right\|_F^2\\ =&\left\|\Pi_{\mathcal{C}}(\boldsymbol{X}_t - \gamma\nabla f_t(\boldsymbol{X}_t)) - \overline{\boldsymbol{X}}\right\|_F^2\\ \le&\left\|\boldsymbol{X}_t - \gamma\nabla f_t(\boldsymbol{X}_t) - \overline{\boldsymbol{X}}\right\|_F^2\\ =&\left\|\boldsymbol{X}_t - \overline{\boldsymbol{X}}\right\|_F^2 + \gamma^2\|\nabla f_t(\boldsymbol{X}_t)\|_F^2 - 2\gamma\langle\nabla f_t(\boldsymbol{X}_t), \boldsymbol{X}_t - \overline{\boldsymbol{X}}\rangle.\end{aligned}$$

Rearranging the terms yields

$$\langle\nabla f_t(\boldsymbol{X}_t), \boldsymbol{X}_t - \overline{\boldsymbol{X}}\rangle \le \frac{\left\|\boldsymbol{X}_t - \overline{\boldsymbol{X}}\right\|_F^2 - \left\|\boldsymbol{X}_{t+1} - \overline{\boldsymbol{X}}\right\|_F^2}{2\gamma} + \frac{\gamma\|\nabla f_t(\boldsymbol{X}_t)\|_F^2}{2}. \tag{7}$$

Accordingly, we can show that

$$
\begin{aligned}
&\mathrm{Regret}_T(\overline{\boldsymbol{X}}) \\
=&\sum_{t=0}^{T-1} f_t(\boldsymbol{X}_t) - \sum_{t=0}^{T-1} f_t(\overline{\boldsymbol{X}}) \\
\leq&\sum_{t=0}^{T-1} \left(\langle \nabla f_t(\boldsymbol{X}_t), \boldsymbol{X}_t - \overline{\boldsymbol{X}} \rangle - \frac{\beta}{2\gamma} \left\| \boldsymbol{X}_t - \overline{\boldsymbol{X}} \right\|_F^2 \right) \\
\leq&-\frac{\beta}{2\gamma} \sum_{t=0}^{T-1} \left\| \boldsymbol{X}_t - \overline{\boldsymbol{X}} \right\|_F^2 + \frac{\left\| \boldsymbol{X}_0 - \overline{\boldsymbol{X}} \right\|_F^2}{2\gamma} - \frac{\left\| \boldsymbol{X}_T - \overline{\boldsymbol{X}} \right\|_F^2}{2\gamma} + \sum_{t=0}^{T-1} \frac{\gamma \left\| \nabla f_t(\boldsymbol{X}_t) \right\|_F^2}{2} \\
\leq&-\frac{\beta}{2\gamma} \sum_{t=0}^{T-1} \left\| \boldsymbol{X}_t - \overline{\boldsymbol{X}} \right\|_F^2 + \frac{\left\| \boldsymbol{X}_0 \right\|_F^2 + \left\| \overline{\boldsymbol{X}} \right\|_F^2}{\gamma} + \sum_{t=0}^{T-1} \frac{\gamma \left\| -\frac{\beta}{\gamma} \boldsymbol{G}_{t+1} + \frac{\beta}{\gamma} \boldsymbol{X}_t \right\|_F^2}{2} \\
\leq&-\frac{\beta}{2\gamma} \sum_{t=0}^{T-1} \left(\frac{\left\| \boldsymbol{X}_t \right\|_F^2}{2} - \left\| \overline{\boldsymbol{X}} \right\|_F^2 \right) + \frac{\left\| \boldsymbol{X}_0 \right\|_F^2 + \left\| \overline{\boldsymbol{X}} \right\|_F^2}{\gamma} + \sum_{t=0}^{T-1} \left(\frac{\beta^2 \left\| \boldsymbol{G}_{t+1} \right\|_F^2}{\gamma} + \frac{\beta^2}{\gamma} \left\| \boldsymbol{X}_t \right\|_F^2 \right) \\
=&\sum_{t=0}^{T-1} \left(\frac{\beta}{\gamma} (\beta - \frac{1}{4}) \left\| \boldsymbol{X}_t \right\|_F^2 + \frac{\beta}{2\gamma} \left\| \overline{\boldsymbol{X}} \right\|_F^2 + \frac{\beta^2}{\gamma} \left\| \boldsymbol{G}_{t+1} \right\|_F^2 \right) + \frac{\left\| \boldsymbol{X}_0 \right\|_F^2 + \left\| \overline{\boldsymbol{X}} \right\|_F^2}{\gamma}.
\end{aligned}
$$

The first inequality is due to the β/γ-strong convexity of $f_t(\cdot)$; the second inequality follows from (7); the third inequality follows from Young's inequality. The last inequality is due to the reverse Young's inequality: $\|\boldsymbol{A} - \boldsymbol{B}\|_F^2 \geq \|\boldsymbol{A}\|_F^2 / 2 - \|\boldsymbol{B}\|_F^2$. Since $\beta \leq 1/8$, then we have

$$
\mathrm{Regret}_T(\overline{\boldsymbol{X}}) \leq \sum_{t=0}^{T-1} \left(-\frac{\beta}{8\gamma} \left\| \boldsymbol{X}_t \right\|_F^2 + \frac{\beta}{2\gamma} \left\| \overline{\boldsymbol{X}} \right\|_F^2 + \frac{\beta^2}{\gamma} \left\| \boldsymbol{G}_{t+1} \right\|_F^2 \right) + \frac{\left\| \boldsymbol{X}_0 \right\|_F^2 + \left\| \overline{\boldsymbol{X}} \right\|_F^2}{\gamma}.
$$

□

B.2 Proof of Lemma 4.2

We first restate Lemma 4.2 as follows.

Lemma B.2. *Let $\gamma = \beta/\eta$ and $\eta \leq 1/\rho$ where $\beta \leq 1/8$, then for any $k \in [K]$, the sequence $\{\boldsymbol{W}_t^{(k)}\}_{t=1}^T$ generated by Algorithm 2 satisfies*

$$
\begin{aligned}
&\mathbb{E}\left[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)}) \right] \\
\leq&-\mathbb{E}\left[\frac{\beta D T}{\gamma} \left\| \overline{\boldsymbol{H}}^{(k)} \right\|_F + \sum_{t=0}^{T-1} \frac{\beta}{8\gamma} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2 \right] + \frac{\beta D \sigma \sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{1}{\gamma} \right) D^2 + \frac{\beta^2 L^2 T}{\gamma} + \frac{r D^2}{\gamma}.
\end{aligned}
$$

Proof. Let us consider the filtration $\mathcal{F}_n = \sigma(\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n)$, where $\sigma(\cdot)$ denotes the generated σ-field. From Assumption 3.4, we can infer that

$$
\boldsymbol{H}_n \coloneqq \mathbb{E}[\boldsymbol{G}_n \mid \mathcal{F}_{n-1}] \in \partial f(\boldsymbol{W}_n).
$$

The weak convexity of $f(\cdot)$ implies that for all $n \geq 1$ and $\eta^{-1} \geq \rho$,

$$
\begin{aligned}
f(\boldsymbol{W}_{n+1}) - f(\boldsymbol{W}_n) \leq& \langle \boldsymbol{H}_{n+1}, \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \rangle + \frac{1}{2\eta} \left\| \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \right\|_F^2 \\
=& \mathbb{E}\left[\langle \boldsymbol{G}_{n+1}, \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \rangle + \frac{1}{2\eta} \left\| \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \right\|_F^2 \mid \mathcal{F}_n \right].
\end{aligned}
$$

The law of total expectation implies that

$$
\mathbb{E}[f(\boldsymbol{W}_{n+1}) - f(\boldsymbol{W}_n)] \leq \mathbb{E}\left[\langle \boldsymbol{G}_{n+1}, \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \rangle + \frac{1}{2\eta} \left\| \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \right\|_F^2 \right].
$$

Fix an arbitrary $k \in [K]$. Recall that $\boldsymbol{W}_t^{(k)} = \boldsymbol{W}_{(k-1)T+t}$ where $t \in \{0\} \cup [T-1]$ and we similarly define the notions of $\boldsymbol{G}_t^{(k)}$, $\boldsymbol{H}_t^{(k)}$ and $\boldsymbol{M}_t^{(k)}$. We can express the previous inequality as

$$\mathbb{E}[f(\boldsymbol{W}_{t+1}^{(k)}) - f(\boldsymbol{W}_t^{(k)})] \leq \mathbb{E}\left[\langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{W}_{t+1}^{(k)} - \boldsymbol{W}_t^{(k)} \rangle + \frac{1}{2\eta} \left\| \boldsymbol{W}_{t+1}^{(k)} - \boldsymbol{W}_t^{(k)} \right\|_F^2\right].$$

Summing the above inequality over $t = 0, \ldots, T-1$ yields that

$$\mathbb{E}[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)})] \leq \mathbb{E}\left[\sum_{t=0}^{T-1} \left(\langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{W}_{t+1}^{(k)} - \boldsymbol{W}_t^{(k)} \rangle + \frac{1}{2\eta} \left\| \boldsymbol{W}_{t+1}^{(k)} - \boldsymbol{W}_t^{(k)} \right\|_F^2\right)\right]. \quad (8)$$

From the update rule of $\boldsymbol{W}_{t+1}^{(k)}$ (step 14 in Algorithm 2), we have

$$\boldsymbol{M}_t^{(k)} = \frac{\boldsymbol{W}_t^{(k)} - \boldsymbol{W}_{t+1}^{(k)}}{\eta},$$

then we can rewrite the inequality (8) as:

$$\mathbb{E}[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)})] \leq \mathbb{E}\left[\sum_{t=0}^{T-1} \left(-\eta \langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{M}_t^{(k)} \rangle + \frac{\eta}{2} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2\right)\right]. \quad (9)$$

If we choose $\beta = \eta\gamma$, and we denote the quadratic function

$$f_t^{(k)}(\boldsymbol{X}) = -\frac{\beta}{\gamma} \langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{X} \rangle + \frac{\beta}{2\gamma} \|\boldsymbol{X}\|_F^2. \quad (10)$$

Since $\beta \leq 1/8$, we can apply Lemma B.1 for the regret analysis with respect to the quadratic function $f_t^{(k)}(\boldsymbol{X})$ by taking $\boldsymbol{X}_t = \boldsymbol{M}_t^{(k)}$ and $\overline{\boldsymbol{X}} = \overline{\boldsymbol{M}}^{(k)}, \mathcal{C} = \{\boldsymbol{X} : \|\boldsymbol{X}\|_2 \leq D\}$:

$$\begin{aligned}
&\mathrm{Regret}_T(\overline{\boldsymbol{M}}) \\
&:= \sum_{t=0}^{T-1} f_t^{(k)}(\boldsymbol{M}_t^{(k)}) - \sum_{t=0}^{T-1} f_t^{(k)}(\overline{\boldsymbol{M}}^{(k)}) \\
&\leq \sum_{t=0}^{T-1} \left(-\frac{\beta}{8\gamma} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2 + \frac{\beta}{2\gamma} \left\| \overline{\boldsymbol{M}}^{(k)} \right\|_F^2 + \frac{\beta^2}{\gamma} \left\| \boldsymbol{G}_{t+1}^{(k)} \right\|_F^2\right) + \frac{\left\| \boldsymbol{M}_0^{(k)} \right\|_F^2 + \left\| \overline{\boldsymbol{M}}^{(k)} \right\|_F^2}{\gamma}.
\end{aligned} \quad (11)$$

Substitute the definition of the quadratic function (10) into (11):

$$\begin{aligned}
&\sum_{t=0}^{T-1} \left(-\frac{\beta}{\gamma} \langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{M}_t^{(k)} \rangle + \frac{\beta}{2\gamma} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2\right) \\
\leq &\sum_{t=0}^{T-1} \left(-\frac{\beta}{\gamma} \langle \boldsymbol{G}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)} \rangle + \frac{\beta}{2\gamma} \left\| \overline{\boldsymbol{M}}^{(k)} \right\|_F^2\right) + \sum_{t=0}^{T-1} \left(-\frac{\beta}{8\gamma} \left\| \boldsymbol{M}_t^{(k)} \right\|_F^2 + \frac{\beta}{2\gamma} \left\| \overline{\boldsymbol{M}}^{(k)} \right\|_F^2 + \frac{\beta^2}{\gamma} \left\| \boldsymbol{G}_{t+1}^{(k)} \right\|_F^2\right) \\
&\quad + \frac{\left\| \boldsymbol{M}_0^{(k)} \right\|_F^2 + \left\| \overline{\boldsymbol{M}}^{(k)} \right\|_F^2}{\gamma}.
\end{aligned}$$

Taking expectations on both sides yields that:

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(-\frac{\beta}{\gamma}\langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{M}_t^{(k)}\rangle + \frac{\beta}{2\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right)\right] \\
\leq &\mathbb{E}\left[\sum_{t=0}^{T-1}\left(-\frac{\beta}{\gamma}\langle \boldsymbol{G}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)}\rangle + \frac{\beta}{2\gamma}\left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F^2\right)\right. \\
&\left.+\sum_{t=0}^{T-1}\left(-\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2 + \frac{\beta}{2\gamma}\left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F^2 + \frac{\beta^2}{\gamma}\left\|\boldsymbol{G}_{t+1}^{(k)}\right\|_F^2\right) + \frac{\left\|\boldsymbol{M}_0^{(k)}\right\|_F^2 + \left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F^2}{\gamma}\right] \\
= &\mathbb{E}\left[-\frac{\beta}{\gamma}\sum_{t=0}^{T-1}\langle \boldsymbol{G}_{t+1}^{(k)} - \boldsymbol{H}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)}\rangle - \frac{\beta}{\gamma}\sum_{t=0}^{T-1}\langle \boldsymbol{H}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)}\rangle\right. \\
&\left.+\sum_{t=0}^{T-1}\left(-\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2 + \frac{\beta}{\gamma}\left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F^2 + \frac{\beta^2}{\gamma}\left\|\boldsymbol{G}_{t+1}^{(k)}\right\|_F^2\right) + \frac{\left\|\boldsymbol{M}_0^{(k)}\right\|_F^2 + \left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F^2}{\gamma}\right].
\end{aligned} \tag{12}
$$

We choose

$$
\overline{\boldsymbol{M}}^{(k)} = D\frac{\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}}{\left\|\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}\right\|_F}, \tag{13}
$$

then we can show that

$$
\left\|\overline{\boldsymbol{M}}^{(k)}\right\|_2 \leq \left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F = D. \tag{14}
$$

In addition, using Jensen's inequality and the fact that $\{\boldsymbol{G}_{t+1}^{(k)} - \boldsymbol{H}_{t+1}^{(k)}\}$ is a martingale-difference sequence with variance bounded by σ^2 (Assumption 3.4):

$$
\begin{aligned}
\mathbb{E}\left[-\frac{\beta}{\gamma}\sum_{t=0}^{T-1}\langle \boldsymbol{G}_{t+1}^{(k)} - \boldsymbol{H}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)}\rangle\right] \leq &\mathbb{E}\left[\frac{\beta}{\gamma}\left\|\sum_{t=0}^{T-1}\left(\boldsymbol{G}_{t+1}^{(k)} - \boldsymbol{H}_{t+1}^{(k)}\right)\right\|_F\left\|\overline{\boldsymbol{M}}^{(k)}\right\|_F\right] \\
\leq &\frac{\beta D\sigma\sqrt{T}}{\gamma},
\end{aligned} \tag{15}
$$

where the last inequality follows from Assumption 3.4. In addition, the definition (13) implies that

$$
\begin{aligned}
\mathbb{E}\left[-\frac{\beta}{\gamma}\sum_{t=0}^{T-1}\langle \boldsymbol{H}_{t+1}^{(k)}, \overline{\boldsymbol{M}}^{(k)}\rangle\right] = &-\mathbb{E}\left[\frac{\beta}{\gamma}\left\langle\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}, \frac{D\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}}{\left\|\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}\right\|_F}\right\rangle\right] \\
= &-\mathbb{E}\left[\frac{\beta D}{\gamma}\left\|\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}\right\|_F\right].
\end{aligned} \tag{16}
$$

Furthermore, Assumption 3.3 implies that $\left\|\boldsymbol{G}_{t+1}^{(k)}\right\|_F \leq L$. Step (13) in Algorithm 2 yields that $\left\|\boldsymbol{M}_0^{(k)}\right\|_F \leq \sqrt{r}D$. Consequently, substituting (14), (15), (16) into (12), we have

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{T-1}\left(-\frac{\beta}{\gamma}\langle \boldsymbol{G}_{t+1}^{(k)}, \boldsymbol{M}_t^{(k)}\rangle + \frac{\beta}{2\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right)\right] \\
\leq &\frac{\beta D\sigma\sqrt{T}}{\gamma} - \mathbb{E}\left[\frac{\beta D}{\gamma}\left\|\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}\right\|_F + \sum_{t=0}^{T-1}\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right] + \left(\frac{\beta TD^2}{\gamma} + \frac{\beta^2L^2T}{\gamma}\right) + \frac{rD^2}{\gamma} + \frac{D^2}{\gamma} \\
= &-\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F + \sum_{t=0}^{T-1}\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right] + \frac{\beta D\sigma\sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{1}{\gamma}\right)D^2 + \frac{\beta^2L^2T}{\gamma} + \frac{rD^2}{\gamma}.
\end{aligned} \tag{17}
$$

where in the last equality follows from the definition $\overline{\boldsymbol{H}}^{(k)} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}$. Recall that $\beta = \eta\gamma$, then combining (9) and (17) yields that

$$\begin{aligned}&\mathbb{E}[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)})]\\ \leq &-\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F + \sum_{t=0}^{T-1}\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right] + \frac{\beta D\sigma\sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma}+\frac{1}{\gamma}\right)D^2 + \frac{\beta^2L^2T}{\gamma} + \frac{rD^2}{\gamma}.\end{aligned}$$

□

Before proceeding, we establish an auxiliary lemma that will be used to control the deviation of the iterates $\boldsymbol{W}_t^{(k)}$ from their average.

Lemma B.3. *Let $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ be a set of matrices, and let $\overline{\boldsymbol{X}} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{X}_i$. Then for all $i \in [n]$,*

$$\left\|\boldsymbol{X}_i - \overline{\boldsymbol{X}}\right\|_F^2 \leq \frac{1}{n}\sum_{i'=1}^{n}\|\boldsymbol{X}_i - \boldsymbol{X}_{i'}\|_F^2 \leq n\sum_{j=1}^{n}\|\Delta_j\|_F^2,$$

where $\Delta_j \coloneqq \boldsymbol{X}_j - \boldsymbol{X}_{j-1}$ and $\boldsymbol{X}_0$ can be chosen arbitrarily.

Proof. Fix any $i \in [n]$, we have

$$\begin{aligned}\left\|\boldsymbol{X}_i - \overline{\boldsymbol{X}}\right\|_F^2 = &\left\|\boldsymbol{X}_i - \frac{1}{n}\sum_{i'=1}^{n}\boldsymbol{X}_{i'}\right\|_F^2 \leq \frac{1}{n}\sum_{i'=1}^{n}\|\boldsymbol{X}_i - \boldsymbol{X}_{i'}\|_F^2\\ =&\frac{1}{n}\sum_{i'=1}^{n}\left\|\sum_{j=i\wedge i'+1}^{i\vee i'}(\boldsymbol{X}_{j-1} - \boldsymbol{X}_j)\right\|_F^2\\ \leq&\frac{1}{n}\sum_{i'=1}^{n}\left(\sum_{j=i\wedge i'+1}^{i\vee i'}\|\Delta_j\|_F\right)^2\\ \leq&\left(\sum_{j=2}^{n}\|\Delta_j\|_F\right)^2 \leq n\sum_{j=1}^{n}\|\Delta_j\|_F^2,\end{aligned} \tag{18}$$

where the first and the last inequalities are due to the Cauchy–Schwarz inequality; The second inequality follows from the triangle inequality. □

B.3 Proof of Theorem 4.3

We now provide the proof of the main convergence theorem of the Musec. We first restate Theorem 4.3 as follows:

Theorem B.4. *Let $\gamma = \beta/\eta$ and $\eta \leq 1/\rho$ where $\beta \leq 1/8$, Then for any $\delta \geq \eta T D\sqrt{r}$, the sequence $\{\overline{\boldsymbol{W}}^{(k)}\}_{k=1}^{K}$ generated by Algorithm 2 satisfies*

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\operatorname{dist}(0, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)}))\right] \leq \frac{\sigma}{\sqrt{T}} + \left(1 + \frac{2r}{\beta T}\right)D + \frac{\beta L^2}{D} + \frac{\gamma\Delta_f}{\beta DTK}.$$

Proof. For any $k \in [K]$, according to Lemma 4.2, one has

$$\begin{aligned}&\mathbb{E}[f(\boldsymbol{W}_T^{(k)}) - f(\boldsymbol{W}_0^{(k)})]\\ \leq &-\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F + \sum_{t=0}^{T-1}\frac{\beta}{8\gamma}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2\right] + \frac{\beta D\sigma\sqrt{T}}{\gamma}\\ &+\left(\frac{\beta T}{\gamma}+\frac{1}{\gamma}\right)D^2 + \frac{\beta^2L^2T}{\gamma} + \frac{rD^2}{\gamma}\\ \leq &-\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F\right] + \frac{\beta D\sigma\sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma}+\frac{2r}{\gamma}\right)D^2 + \frac{\beta^2L^2T}{\gamma}.\end{aligned}$$

By the definition that $\boldsymbol{W}_T^{(k)} = \boldsymbol{W}_0^{(k+1)}$, we obtain

$$\begin{aligned}&\mathbb{E}[f(\boldsymbol{W}_0^{(k+1)}) - f(\boldsymbol{W}_0^{(k)})]\\ \leq &-\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F\right] + \frac{\beta D\sigma\sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{2r}{\gamma}\right)D^2 + \frac{\beta^2 L^2 T}{\gamma}.\end{aligned}$$

Rearranging the terms on both sides of the inequality, we have

$$\mathbb{E}\left[\frac{\beta DT}{\gamma}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F\right] \leq \frac{\beta D\sigma\sqrt{T}}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{2r}{\gamma}\right)D^2 + \frac{\beta^2 L^2 T}{\gamma} + \mathbb{E}[f(\boldsymbol{W}_0^{(k)}) - f(\boldsymbol{W}_0^{(k+1)})].$$

By telescoping both sides through $k \in [K]$, we have

$$\begin{aligned}\mathbb{E}\left[\frac{\beta DT}{\gamma}\sum_{k=1}^{K}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F\right] \leq &\frac{\beta D\sigma\sqrt{T}K}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{2r}{\gamma}\right)D^2K + \frac{\beta^2 L^2 TK}{\gamma} + \mathbb{E}[f(\boldsymbol{W}_0^{(0)}) - f(\boldsymbol{W}_0^{(K+1)})]\\ \leq &\frac{\beta D\sigma\sqrt{T}K}{\gamma} + \left(\frac{\beta T}{\gamma} + \frac{2r}{\gamma}\right)D^2K + \frac{\beta^2 L^2 TK}{\gamma} + \Delta_f,\end{aligned}$$

where the last inequality follows from the definition of Δ_f. Dividing both sides by $\beta DTK/\gamma$ yields that

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F\right] \leq \frac{\sigma}{\sqrt{T}} + \left(1 + \frac{2r}{\beta T}\right)D + \frac{\beta L^2}{D} + \frac{\gamma\Delta_f}{\beta DTK}.$$

To translate the bound on $\left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F$ into a Goldstein-stationarity bound at $\overline{\boldsymbol{W}}^{(k)}$, we need every iterate $\boldsymbol{W}_t^{(k)}$ to lie within the δ-ball of $\overline{\boldsymbol{W}}^{(k)}$. Applying Lemma B.3, for any $t \in [T]$, we can show that

$$\left\|\boldsymbol{W}_t^{(k)} - \overline{\boldsymbol{W}}^{(k)}\right\|_F \leq \sqrt{T\sum_{t=1}^{T}\left\|\boldsymbol{W}_t^{(k)} - \boldsymbol{W}_{t-1}^{(k)}\right\|_F^2} \leq \sqrt{T\eta^2\sum_{t=0}^{T-1}\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2} \leq \eta TD\sqrt{r} \leq \delta,$$

where the second inequality follows from the step (14) in Algorithm 2 and the third inequality is due to $\left\|\boldsymbol{M}_t^{(k)}\right\|_F^2 \leq rD^2$. It implies that

$$\operatorname{dist}(\mathbf{0}, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)})) \leq \left\|\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{H}_t^{(k)}\right\|_F = \left\|\overline{\boldsymbol{H}}^{(k)}\right\|_F.$$

Consequently, we obtain

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\operatorname{dist}(\mathbf{0}, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)}))\right] \leq \frac{\sigma}{\sqrt{T}} + \left(1 + \frac{2r}{\beta T}\right)D + \frac{\beta L^2}{D} + \frac{\gamma\Delta_f}{\beta DTK}.$$

□

B.4 Proof of Corollary 4.4

We restate the complexity bound of Musec as follows.

Corollary B.5. *By choosing the parameters as*

$$\begin{aligned}&T = (\delta N)^{2/3}, \quad K = \delta^{-2/3}N^{1/3}, \quad D = (\delta N)^{-1/3},\\ &\beta = \frac{r^{1/2}}{(\delta N)^{2/3}}, \quad \gamma = \frac{r}{\delta^{4/3}N^{1/3}}, \quad \eta = \frac{\delta^{2/3}}{\sqrt{r}N^{1/3}},\end{aligned}$$

then it takes at most

$$N = \mathcal{O}\left(\frac{r^{3/4}}{\delta} + \frac{\delta^2\rho^3}{r^{3/2}} + \frac{r^{3/2}(\sigma^3 + L^6 + \Delta_f^3)}{\delta\epsilon^3}\right)$$

to obtain a (δ, ϵ)-Goldstein stationary point.

Proof. The theorem imposes four constraints on the parameters:

$$\eta TD\sqrt{r} \le \delta, \quad \beta = \eta\gamma, \quad \beta \le \frac{1}{8}, \quad \eta^{-1} \ge \rho$$

We first specify the parameters, then verify that each constraint is satisfied under a mild lower bound on N, and finally translate the theorem's bound into the stated oracle complexity.

Step 1: Parameter choices. Set

$$\begin{aligned} &T = (\delta N)^{2/3}, \quad K = \frac{N}{T} = \delta^{-2/3}N^{1/3}, \quad D = (\delta N)^{-1/3}, \\ &\beta = \frac{r^{1/2}}{(\delta N)^{2/3}}, \quad \gamma = \frac{r}{\delta^{4/3}N^{1/3}}, \quad \eta = \frac{\delta^{2/3}}{\sqrt{r}N^{1/3}} \end{aligned} \tag{19}$$

Step 2: Verifying the constraints. A direct computation gives

$$\eta\gamma = \frac{\delta^{2/3}}{\sqrt{r}N^{1/3}} \cdot \frac{r}{\delta^{4/3}N^{1/3}} = \beta,$$

so $\beta = \eta\gamma$ holds by construction. For the step-size constraint,

$$\eta TD\sqrt{r} = \frac{\delta^{2/3}}{\sqrt{r}N^{1/3}}(\delta N)^{1/3}\sqrt{r} \le \delta,$$

so $\eta TD\sqrt{r} \le \delta$ also holds. The remaining two constraints $\beta \le \frac{1}{8}$ and $\eta^{-1} \ge \rho$ translate into lower bounds on N:

$$N \ge \mathcal{O}\left(\frac{r^{3/4}}{\delta}\right), \qquad N \ge \frac{\rho^3\delta^2}{r^{3/2}}.$$

Step 3: Evaluating the theorem's bound. By the theorem 4.3 implies that

$$\begin{aligned} \mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\operatorname{dist}(\mathbf{0}, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)}))\right] \le& \frac{\sigma}{\sqrt{T}} + \left(1 + \frac{2r}{\beta T}\right)D + \frac{\beta L^2}{D} + \frac{\Delta_f\gamma}{\beta DTK} \\ \le& \frac{\sigma}{(\delta N)^{\frac{1}{3}}} + \frac{1}{(\delta N)^{\frac{1}{3}}} + \frac{2r^{\frac{1}{2}}}{(\delta N)^{\frac{1}{3}}} + \frac{r^{\frac{1}{2}}L^2}{(\delta N)^{\frac{1}{3}}} + \frac{\Delta_f r^{\frac{1}{2}}}{(\delta N)^{\frac{1}{3}}}, \end{aligned}$$

where the last inequality follows from the parameter choices in (19). To obtain (δ, ϵ)-Goldstein stationary point such that it satisfies

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\operatorname{dist}(\mathbf{0}, \partial_\delta f(\overline{\boldsymbol{W}}^{(k)}))\right] \le \epsilon,$$

it takes at most

$$N = \mathcal{O}\left(\frac{r^{\frac{3}{4}}}{\delta} + \frac{\delta^2\rho^3}{r^{\frac{3}{2}}} + \frac{r^{\frac{3}{2}}(\sigma^3 + L^6 + \Delta_f^3)}{\delta\epsilon^3}\right)$$

stochastic gradient oracle queries. □

C Detailed Algorithm of Soft Musec

In Section 5, we introduced Soft Musec, which replaces exact SVD-based spectral clipping with a smooth saturation function approximated via coupled Newton–Schulz iterations. Here we provide the full algorithmic details.

Recall from Section 5 that the soft spectral clipping operator is given by

$$H(\boldsymbol{M}, D) = D(\boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I})^{-1/2}\boldsymbol{M},$$

Algorithm 3: Soft Spectral Clipping (SSC)

Input: Matrix $\boldsymbol{M} \in \mathbb{R}^{m\times n}$, clipping threshold $D > 0$, number of Newton–Schulz iterations K.
$\boldsymbol{A} = \boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I}$
$\alpha = \|\boldsymbol{A}\|_F$
$\boldsymbol{Y}_0 = \boldsymbol{A}/\alpha$
$\boldsymbol{Z}_0 = \boldsymbol{I}$
for $k = 0, 1, \ldots, K-1$ **do**
  $\boldsymbol{T}_k = \frac{1}{2}(3\boldsymbol{I} - \boldsymbol{Z}_k\boldsymbol{Y}_k)$
  $\boldsymbol{Y}_{k+1} = \boldsymbol{Y}_k\boldsymbol{T}_k$
  $\boldsymbol{Z}_{k+1} = \boldsymbol{T}_k\boldsymbol{Z}_k$
end
Return: $D \cdot (\boldsymbol{Z}_K/\sqrt{\alpha}) \cdot \boldsymbol{M}$.

Algorithm 4: Soft Musec

Input: Initial point $\boldsymbol{W}_0$, momentum parameter $\beta \in [0, 1)$, learning rate η, clipping threshold $D > 0$, positive integers K and T, number of Newton–Schultz steps K_{ns}
$N = K \times T$
for $n = 0, 1, \ldots, N-1$ **do**
  Sample $\xi_n \sim \mathcal{P}$
  $\boldsymbol{G}_n = G(\boldsymbol{W}_n; \xi_n)$
  if $n = 0$ **then**
    $\widehat{\boldsymbol{M}}_0 = \boldsymbol{G}_0$
  else
    $\widehat{\boldsymbol{M}}_n = (1-\beta)\boldsymbol{M}_{n-1} + \beta\boldsymbol{G}_n$
  end
  $\boldsymbol{M}_n = \mathrm{SSC}(\widehat{\boldsymbol{M}}_n, D, K_{\text{ns}})$ `// Algorithm 3`
  $\boldsymbol{W}_{n+1} = \boldsymbol{W}_n - \eta\boldsymbol{M}_n$
end
Set $\overline{\boldsymbol{W}}^{(k)} = \frac{1}{T}\sum_{t=0}^{T-1} \boldsymbol{W}_t^{(k)}$ where $\boldsymbol{W}_t^{(k)} = \boldsymbol{W}_{(k-1)T+t}$ for $\forall k \in [K]$
Return: $\overline{\boldsymbol{W}} \sim \mathrm{Uniform}(\{\overline{\boldsymbol{W}}^{(k)} : k \in [K]\})$.

where $\boldsymbol{M} \in \mathbb{R}^{m\times n}$ with $m \le n$ w.l.o.g. We approximate the matrix inverse square root

$$(\boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I})^{-1/2}$$

using coupled Newton–Schulz iterations (Higham, 2008, Chapter 6), which involve only matrix–matrix multiplications and are therefore well-suited to GPU computation. The complete procedure is presented in Algorithm 3. The normalization by $\left\|\boldsymbol{M}\boldsymbol{M}^\top + D^2\boldsymbol{I}\right\|_F$ ensures that the initial iterate $\boldsymbol{Y}_0$ has unit Frobenius norm, placing the eigenvalues in a range where the Newton–Schulz iteration converges. The addition of $D^2\boldsymbol{I}$ shifts all eigenvalues away from zero, ensuring numerical stability.

We present the full Soft Musec procedure in Algorithm 4, which integrates the soft spectral clipping subroutine (Algorithm 3) into the Musec framework by replacing the exact SVD and hard clipping in Steps 11–14 of Algorithm 2.

D Comparison with SPECTRA

We provide a detailed comparison of the Musec and SPECTRA update rules to clarify their algorithmic differences.

Given a matrix $\boldsymbol{M}$ with reduced SVD $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$, we define the spectral clipping operator $\mathbf{Clip}_{\mathrm{sc}}(\boldsymbol{M}) = \boldsymbol{U}\mathbf{Clip}(\boldsymbol{S}, D)\boldsymbol{V}^\top$. Both Musec and SPECTRA employ this operator, but they differ in how the clipped output interacts with the momentum state.

Musec maintains a clipped momentum buffer and computes:

$$\begin{cases} \widehat{\boldsymbol{M}}_n = (1-\beta)\boldsymbol{M}_{n-1} + \beta\boldsymbol{G}_n, \\ \boldsymbol{M}_n = \mathbf{Clip}_{\mathrm{sc}}(\widehat{\boldsymbol{M}}_n, D), \\ \boldsymbol{W}_{n+1} = \boldsymbol{W}_n - \eta\boldsymbol{M}_n, \end{cases}$$

Crucially, the clipped momentum $\boldsymbol{M}_n$ is propagated to the next iteration. This ensures that the momentum state is always spectrally bounded: $\|\boldsymbol{M}_n\| \le D$ at every iteration.

Ignoring weight decay, the SPECTRA update (Jiang et al., 2026b, Eq. (4)) maintains an unclipped momentum buffer and applies spectral clipping only when computing the parameter update:

$$\begin{cases} \boldsymbol{M}_n = (1-\beta)\boldsymbol{M}_{n-1} + \beta\boldsymbol{G}_n. \\ \boldsymbol{W}_{n+1} = \boldsymbol{W}_n - \eta\mathbf{Clip}_{\mathrm{sc}}(\boldsymbol{M}_n, D), \end{cases}$$

Here, the unclipped momentum $\boldsymbol{M}_n$ is carried forward to the next iteration. Although the weight update is spectrally clipped, the momentum state itself is unconstrained.

Key difference. The distinction lies in whether spectral clipping is integrated into the momentum recurrence. In Musec, the momentum state satisfies $\|\boldsymbol{M}_n\| \le D$ at every iteration, so the input to the next EMA step is always spectrally bounded. In SPECTRA, the momentum state is unconstrained: $\|\boldsymbol{M}_n\|$ can grow well beyond D over successive iterations, as clipping is only applied to the output update and does not feed back into the recurrence. Consequently, SPECTRA's momentum can retain large spectral components accumulated over many past iterations, even when recent gradients are small in those directions. Musec's feedback mechanism prevents this accumulation, providing tighter spectral control over the optimization trajectory.

E Experimental Details

This section provides the detailed architecture, training, and hyperparameter configurations for our NanoGPT experiments.

E.1 Architecture Configurations

Table 1: Architecture specifications across model scales.

	Small	**Medium**	**Wide**
Layers	11	16	16
Attention heads	6	8	16
Head dimension	128	128	128
Model dimension	768	1,024	2,048
MLP hidden dimension	3,072	4,096	8,192
Max sequence length	2,048	4,096	4,096
Core parameters	∼491M	∼613M	∼1.63B

In our experiments, we evaluate three model configurations, including NanoGPT-small, NanoGPT-Medium, and NanoGPT-Wide. All three model configurations follow a decoder-only Transformer architecture based on the modded-nanogpt codebase.[2] The architecture incorporates several modern modifications on top of the standard GPT-2 backbone, including RMSNorm (Zhang & Sennrich, 2019), squared ReLU activations (So et al., 2021), Rotary Position Embeddings (Su et al., 2024), sliding-window causal attention via Flash Attention 3 (Shah et al., 2024), and multi-token prediction as an auxiliary training objective. We refer to the codebase for full architectural details. The three configurations differ primarily in their width scaling. Table 1 summarizes their specifications.

[2] `https://github.com/KellerJordan/modded-nanogpt`

E.2 Training Configuration

Table 2: Training configuration across model scales.

	Small	**Medium**	**Wide**
Total steps	1,390	4,740	8,040
Final batch size (tokens)	393K	524K	1,049K
Precision	bfloat16	bfloat16	bfloat16
Hardware	H800	H800	H800

All models are trained using the GPT-2 BPE tokenizer in bfloat16 precision on NVIDIA H800 GPUs. We train all configurations on FineWeb (Penedo et al., 2024), OpenWebText (Gokaslan & Cohen, 2019), and C4 (Raffel et al., 2020). Each configuration employs a multi-phase training schedule that jointly ramps up batch size and learning rate. The learning rate follows a warmup-then-decay schedule. Table 2 summarizes the key training parameters. Full schedule details are available in the modded-nanogpt codebase.

E.3 Hyperparameters

Table 3: Selected weight decay λ and clipping threshold D for each method and learning rate on NanoGPT-Small. Each entry corresponds to the best validation loss over the sweep. Soft Musec and SPECTRA share the same hyperparameter.

	Muon			**MuonClip**			**Soft Musec / SPECTRA**			
LR	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	D
0.01	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.5
0.023	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.5
0.05	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.1
0.1	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.1
0.2	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.1
0.5	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.1

Table 4: Selected weight decay λ and clipping threshold D for each method and learning rate on NanoGPT-Medium. Each entry corresponds to the best validation loss over the sweep. Soft Musec and SPECTRA share the same hyperparameter.

	Muon			**MuonClip**			**Soft Musec / SPECTRA**			
LR	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	D
0.01	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.25
0.015	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.25
0.05	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.1	1.2	1.2	0.3	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.2	1.2	1.2	0.3	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.3	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.05
0.5	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.05
0.8	0.3	0.3	0.1	0.3	0.3	0.1	0.3	0.3	0.1	0.05

For all methods, the respective optimizer is applied to attention and MLP projection matrices, while AdamW is used for embeddings, the language-model head, and scalar parameters. AdamW hyperparameters are held fixed across all methods. For each method, model configuration, and learning rate, we tune the weight decay over $\{0.096, 0.3, 0.6, 1.2\}$. For Soft Musec and SPECTRA, we additionally tune the clipping threshold D over $\{0.05, 0.125, 0.25, 0.5, 0.75, 1.0\}$. We report the best validation loss over these sweeps. Tables 3, 4, and 5 list the selected hyperparameters for NanoGPT-Small, NanoGPT-Medium, and NanoGPT-Wide, respectively.

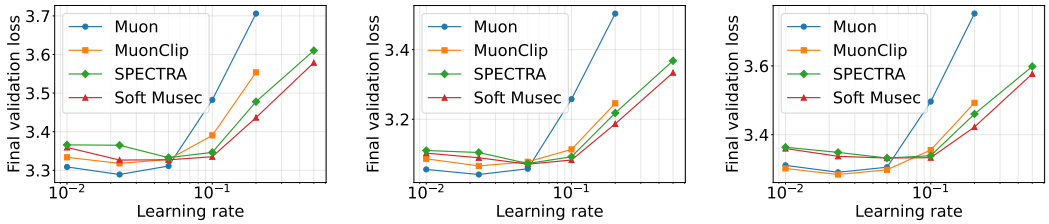
Table 5: Selected weight decay λ and clipping threshold D for each method and learning rate on NanoGPT-Wide. Each entry corresponds to the best validation loss over the sweep. Soft Musec and SPECTRA share the same hyperparameter.

	Muon			**MuonClip**			**Soft Musec / SPECTRA**			
LR	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	λ_{QK}	λ_{VO}	λ_{MLP}	D
0.01	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.25
0.05	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.1	1.2	1.2	0.3	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.2	1.2	1.2	0.3	1.2	1.2	1.2	1.2	1.2	0.3	0.05
0.3	1.2	1.2	0.3	1.2	1.2	0.3	1.2	1.2	0.3	0.05
0.5	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.05

F Additional Experimental Results

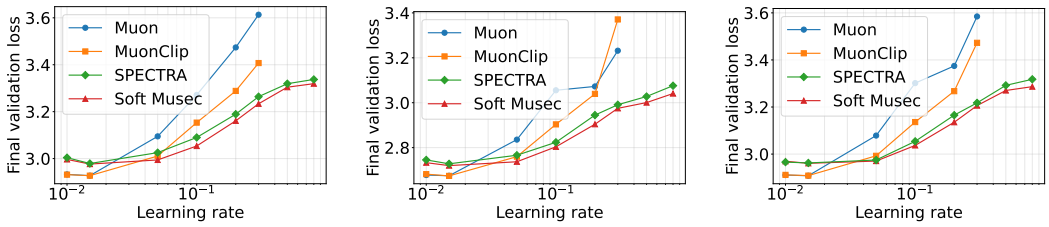
This section presents additional experimental results comparing Soft Musec against baseline methods on NanoGPT training.

F.1 Learning Rate Sweep



(a) FineWeb (b) OpenWebText (c) C4

Figure 4: Validation loss versus learning rate for NanoGPT-Small across three datasets.



(a) FineWeb (b) OpenWebText (c) C4

Figure 5: Validation loss versus learning rate for NanoGPT-Medium across three datasets.

Figures 4–5 present the complete learning rate sweep results across all model configurations and datasets; runs that diverge during training are omitted. For NanoGPT-Small (Figure 4), all methods achieve comparable validation loss at small learning rates. As the learning rate increases, Muon degrades most rapidly, followed by MuonClip, while Soft Musec and SPECTRA maintain lower validation loss across the full range. For NanoGPT-Medium (Figure 5), the advantage of spectral clipping methods becomes more pronounced: at learning rates $\eta \geq 0.1$, both Soft Musec and SPECTRA substantially outperform Muon and MuonClip, with Soft Musec achieving the best or near-best validation loss at the largest learning rates across all three datasets.

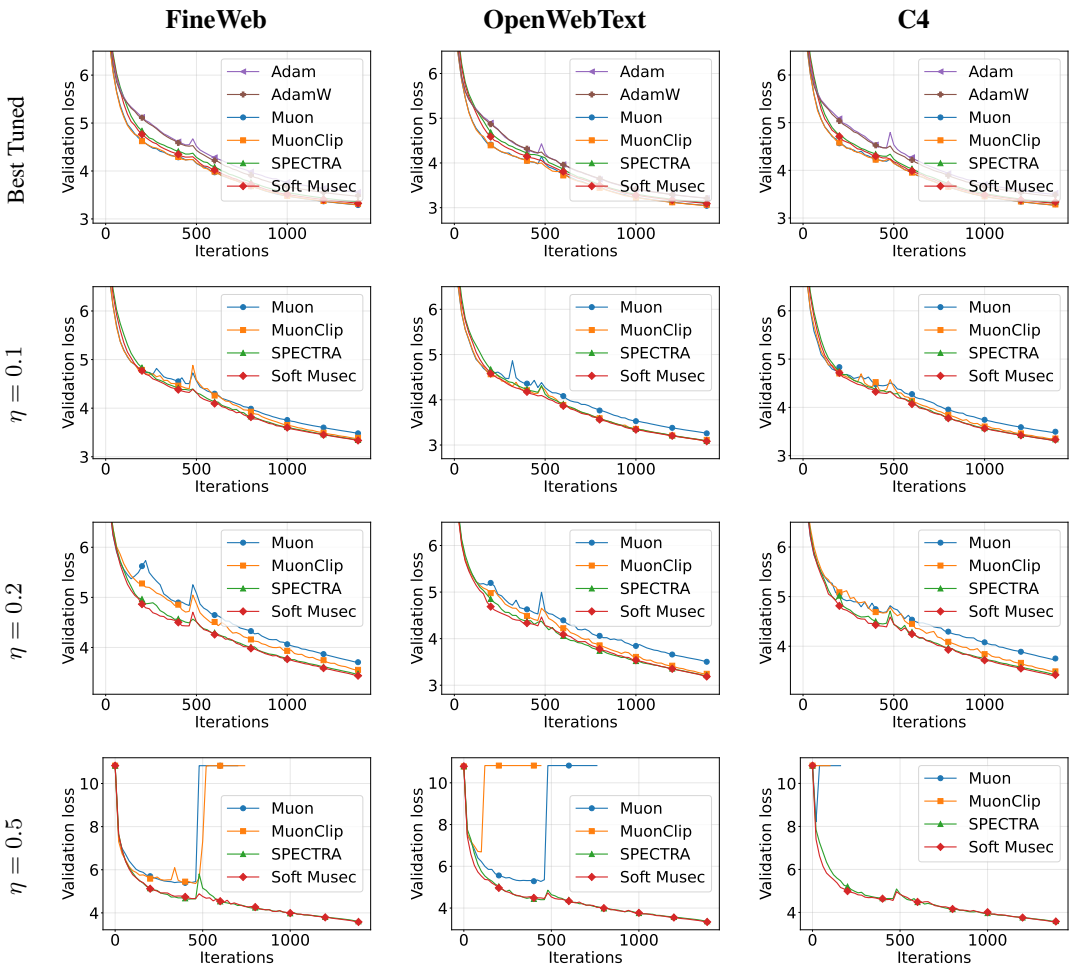


Figure 6: Training loss over steps for NanoGPT-Small. Rows: learning rates; columns: datasets. Only learning rates where methods exhibit meaningfully different behavior are shown. NaN values within runs are omitted.

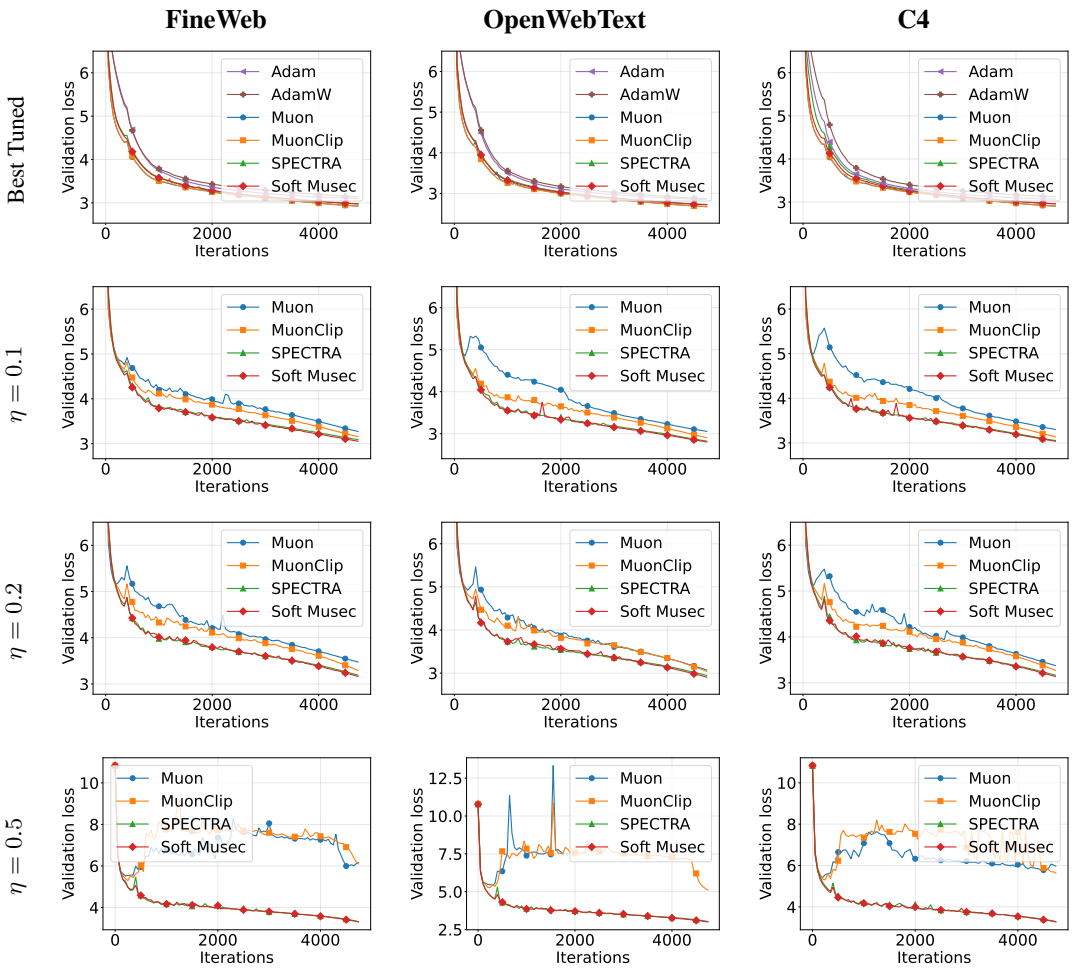


Figure 7: Training loss over steps for NanoGPT-Medium. Rows: learning rates; columns: datasets. Only learning rates where methods exhibit meaningfully different behavior are shown. NaN values within runs are omitted.

F.2 Additional Training Dynamics

Figures 6–8 present training loss curves across all model configurations, datasets, and selected learning rates. At the best-tuned learning rate, all Muon-type optimizers converge to lower validation loss than Adam and AdamW. As the learning rate increases, the stability gap between methods widens, and this effect is amplified at larger model scales. For NanoGPT-Small, instability only manifests at the largest learning rate ($\eta = 0.5$), where Muon and MuonClip diverge on some datasets while Soft Musec and SPECTRA remain stable. For NanoGPT-Medium and NanoGPT-Wide, Muon and MuonClip exhibit increasingly severe instability starting from $\eta = 0.1$, with large loss spikes and sustained oscillations. Soft Musec and SPECTRA both maintain stable convergence across all configurations, with Soft Musec achieving comparable or slightly smoother training dynamics.

F.3 Effective Rank Analysis

To directly validate our theoretical motivation, we track the effective rank of the update matrices across QK projections, VO projections, and MLP weights during NanoGPT-Medium training on FineWeb at $\eta = 0.5$.

Let us first recall the definition of the effective rank (Roy & Vetterli, 2007; Garrido et al., 2023) of a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$, where $r = \min(m, n)$.

We define the singular value distribution as

$$p_i = \frac{\sigma_i}{\sum_{j=1}^{r} \sigma_j}, \qquad i \in [r]$$

so that $\sum_i p_i = 1$. We next compute the Shannon entropy of this distribution as

$$H(p_1, \ldots, p_r) = -\sum_{i=1}^{r} p_i \log(p_i).$$

The effective rank of the matrix $\boldsymbol{A}$, denoted $\mathrm{erank}(\boldsymbol{A})$, is defined as

$$\mathrm{erank}(\boldsymbol{A}) = \exp(H(p_1, \ldots, p_r)).$$

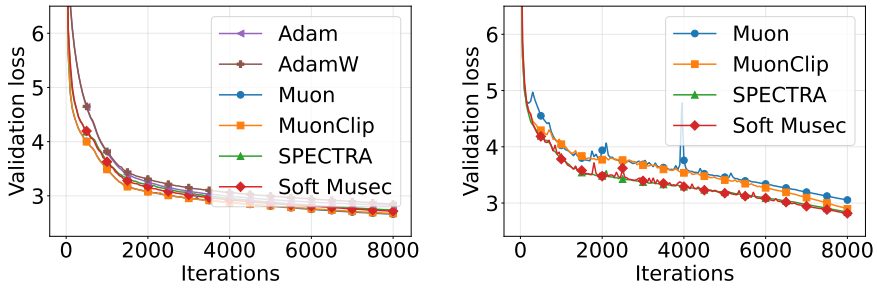
The complete results are shown in Figure 9. Muon exhibits near-full effective rank across all components, confirming that its polar step discards all spectral magnitude information. MuonClip does not meaningfully reduce the effective rank of the update, as it targets the weight matrices rather than the update itself. Both SPECTRA and Soft Musec achieve substantially lower effective rank than Muon, consistent with the effect of spectral clipping. Notably, as training progresses, Soft Musec maintains lower effective rank than SPECTRA across all three component types. This is consistent with the momentum feedback mechanism in Musec: by propagating the clipped momentum, spectral components are continuously regulated rather than allowed to accumulate in an unclipped buffer, as discussed in Section 4.

F.4 Computational Overhead

We compare the average wall-clock time per training step across all methods on FineWeb, with results shown in Figure 10. Across all three model configurations, Soft Musec introduces negligible computational overhead compared to Muon. For NanoGPT-Small, all methods are within 3% of each other (1021–1049 ms). For NanoGPT-Medium, all Muon-type methods achieve similar wall-clock time (2550–2692 ms), with MuonClip being the slowest. For NanoGPT-Wide, all methods remain within 2% (15645–15977 ms). These results confirm that the spectral clipping operation in Soft Musec introduces minimal computational overhead relative to Muon, as both methods rely on Newton–Schulz iterations of similar complexity.

F.5 Ablation Studies

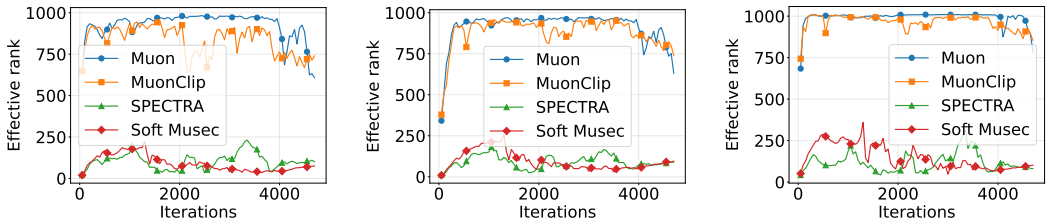
In this subsection, we ablate two key hyperparameters of Soft Musec: the clipping threshold D and the number of Newton–Schulz iterations.



(a) Best Tuned (b) $\eta = 0.1$

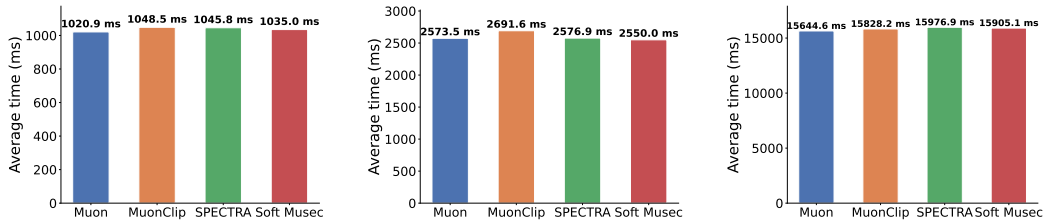
(c) $\eta = 0.2$ (d) $\eta = 0.5$

Figure 8: Training loss over steps for NanoGPT-Wide on FineWeb dataset. Only learning rates where methods exhibit meaningfully different behavior are shown. NaN values within runs are omitted.



(a) QK projections (b) VO projections (c) MLP

Figure 9: Effective rank of update matrices over training steps for NanoGPT-Medium on FineWeb.



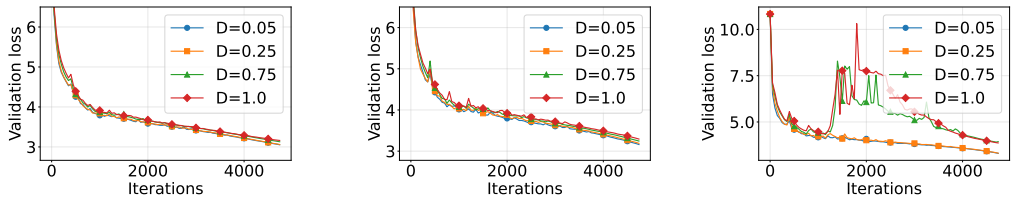
(a) NanoGPT-Small (b) NanoGPT-Medium (c) NanoGPT-Wide

Figure 10: Average wall clock time comparison of methods for training FineWeb.

F.5.1 Sensitivity to Clipping Threshold

We study the sensitivity of Soft Musec to the clipping threshold D on NanoGPT-Medium trained on FineWeb across three learning rates ($\eta \in \{0.1, 0.2, 0.5\}$), with results shown in Figure 11.

At moderate learning rates ($\eta = 0.1$ and $\eta = 0.2$), Soft Musec is robust to the choice of D: all values from $D = 0.05$ to $D = 1.0$ converge to comparable validation loss with similar training dynamics. At the largest learning rate ($\eta = 0.5$), the choice of D becomes more important. Smaller thresholds ($D = 0.05$ and $D = 0.25$) maintain smooth convergence, while larger thresholds ($D = 0.75$ and $D = 1.0$) exhibit loss spikes in the later stages of training. Notably, even with the largest clipping threshold ($D = 1.0$), Soft Musec still converges — in contrast to Muon and MuonClip, which diverge entirely at this learning rate (Figure 7). This demonstrates that spectral clipping provides a meaningful stability benefit regardless of the threshold, while a well-tuned D further improves training smoothness.



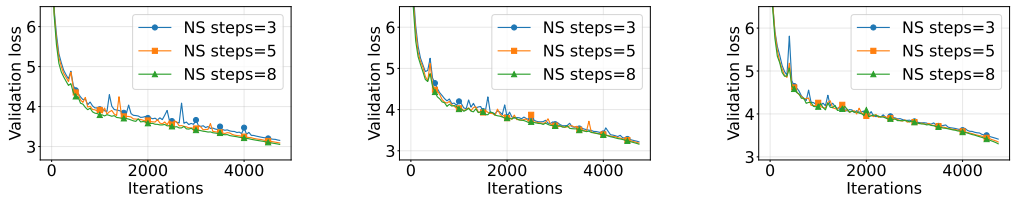
(a) $\eta = 0.1$ (b) $\eta = 0.2$ (c) $\eta = 0.5$

Figure 11: Effect of clipping threshold on Soft Musec training stability across learning rates.

F.5.2 Number of Newton-Schulz Iterations

We study the sensitivity of Soft Musec to the number of Newton–Schulz iterations on NanoGPT-Medium trained on FineWeb across three learning rates ($\eta \in \{0.1, 0.2, 0.5\}$), with the clipping threshold fixed at $D = 0.05$. Results are shown in Figure 12.

Across all learning rates, Soft Musec is largely insensitive to the number of Newton–Schulz iterations. More iterations yield slightly better validation loss, consistent with a more accurate approximation of the spectral clipping operator. However, the differences are marginal, indicating that even a coarse approximation with three iterations provides most of the stability benefit. This robustness stems from the well-conditioned nature of the matrix $\widehat{\boldsymbol{M}}\widehat{\boldsymbol{M}}^\top + D^2\boldsymbol{I}$, whose eigenvalues are bounded below by D^2, enabling rapid convergence of the iteration. In practice, five iterations offer a good balance between approximation quality and computational cost.



(a) $\eta = 0.1$ (b) $\eta = 0.2$ (c) $\eta = 0.5$

Figure 12: Effect of Newton–Schultz steps on Soft Musec training stability across learning rates.