ADAPTIVE OPTIMIZATION VIA MOMENTUM ON VARIANCE-NORMALIZED GRADIENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce MVN-Grad (*Momentum on Variance-Normalized Gradients*), an Adam-style optimizer that improves stability and performance by combining two complementary ideas: variance-based normalization and momentum applied *after* normalization. MVN-Grad scales each coordinate by an exponential moving average of gradient uncertainty and applies momentum to the resulting normalized gradients, removing the cross-time coupling between stale momentum and a stochastic normalizer present in standard Adam-type updates. We prove that, when the mean estimate tracks the conditional mean exactly and the noise is symmetric, this decoupling yields no larger one-step conditional update variance than momentum–then–normalize variance methods, and that, as in LaProp, an isolated spike is normalized before it enters the momentum buffer. In low-variance regimes, we further show that variance normalization avoids sign-type collapse of second-moment scaling in idealized limiting dynamics. Beyond these comparisons, we prove a general nonconvex convergence guarantee for MVN-Grad under bounded-gradient stochastic assumptions. On CIFAR-100 and GPT-style language modeling with models up to 350M parameters, MVN-Grad matches or improves on Adam, AdaBelief, and LaProp, with improved robustness in several evaluated hyperparameter sweeps at the cost of one additional optimizer-state tensor.

1 INTRODUCTION

Adaptive gradient methods such as Adam (Kingma & Ba, 2014) are widely used due to their empirical robustness and ease of hyperparameter tuning. Predecessors like AdaGrad (Duchi et al., 2011) and RMSProp (Hinton et al., 2012) pioneered the use of coordinate-wise preconditioning based on running statistics. Adam builds on these foundations by combining an exponential moving average (EMA) of gradients with per-coordinate normalization based on the second moment, enabling rapid convergence across diverse architectures. Despite their success, a growing body of work has identified failure modes and generalization limitations of Adam-type methods in stochastic settings (Wilson et al., 2017; Reddi et al., 2019; Chen et al., 2018b). We identify two structural limitations in the Adam design that contribute to these failures: temporal coupling and suboptimal scaling.

First, a critical weakness lies in Adam's standard *momentum-then-normalize* ordering, which creates a stochastic coupling between the numerator and denominator. While both terms utilize EMAs, their interaction induces a temporal mismatch. The momentum term m_t acts as a heavy memory buffer for *past* gradients, whereas the adaptive denominator is continuously modulated by the *current* stochastic gradient g_t. This creates a risk of "cross-time" instability: if the normalizer transiently dips—for instance, during a mini-batch with low gradient magnitude—while the momentum buffer still retains a large value from a past outlier, the update can become disproportionately sensitive to the current normalizer. In regimes where the denominator forgets faster than the numerator, this carry-over can lead to spike-dependent responses. This coupling amplifies stale gradients and distorts the update direction, a failure mode that has also been highlighted in some recent work (Ziyin et al., 2020; Zhou et al., 2018; Zhuang et al., 2021).

Second, reliance on the *uncentered second moment* for normalization can lead to ineffective scaling, as $\mathbb{E}[g_t^2] = (\mathbb{E}[g_t])^2 + \mathrm{Var}(g_t)$ mixes signal and noise. In low-variance regimes where the signal dominates, the normalizer tracks the squared gradient magnitude, causing the adaptive term to scale

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.

the gradient by roughly its own size ($g_t/\sqrt{g_t^2} \approx \mathrm{sign}(g_t)$) and degenerate into a coordinate-wise sign operation. This sign-like behavior suppresses coordinate-wise gradient-magnitude information and could slow convergence in high-signal regimes (Balles & Hennig, 2018; Bernstein et al., 2018).

Past attempts to improve Adam tackle these axes individually, trading one limitation for another. To resolve the temporal mismatch, LaProp (Ziyin et al., 2020) applies momentum to the *normalized* gradient, while AdaShift (Zhou et al., 2018) decorrelates numerator and denominator via temporally shifted statistics. Although these methods decouple history from current noise, they retain the second-moment normalizer and remain susceptible to sign collapse. Conversely, AdaBelief (Zhuang et al., 2020) adopts variance-based scaling but ignores the ordering flaw; because variance estimates are smaller and more volatile than second moments, the standard ordering can make them less stable than Adam (Yuan & Gao, 2020). Consequently, the design space remains fragmented, forcing a choice between structural stability and adaptive sensitivity.

Contributions. We introduce MVN-Grad (Momentum on Variance-Normalized Gradients), which addresses two key limitations via a normalize-then-momentum structure and variance-based normalization. Applying momentum *after* normalization removes stochastic coupling and improves stability when the normalizer fluctuates, while replacing the second moment with a variance estimate avoids sign collapse and preserves gradient scale in high-signal regimes. Crucially, these two axes are *orthogonal*: prior methods fix one flaw while leaving the other unresolved. LaProp and AdaShift correct the ordering but retain second-moment normalization (and thus sign collapse); AdaBelief adopts variance-based scaling but retains the momentum-then-normalize ordering (and thus stochastic coupling). MVN-Grad occupies the variance-normalized, normalize-then-momentum cell of this design space; a closely related update appears in centered RMSProp (Graves, 2013). Unlike that formulation, MVN-Grad uses an EMA of squared residuals around a potentially differently decayed mean estimate and develops a dedicated theoretical and empirical analysis of this construction.

- **Theoretical Stability Analysis.** Under exact mean tracking and conditionally symmetric noise, we prove that normalize–then–momentum yields no larger—and under nondegeneracy, strictly smaller—one-step conditional update variance than AdaBelief. Proposition 3.2 extends the comparison to imperfect mean tracking. We also characterize spike carry-over, showing that MVN-Grad has a bounded response to an isolated spike under the stated conditions.
- **High-Signal Convergence.** We show that second-moment normalization can collapse to sign-like updates in low-noise regimes, while variance normalization preserves gradient magnitudes when its variance estimate is stable. In idealized dynamics, this yields an $O(1/T)$ average squared-gradient bound for fixed variance normalization and an explicit dimension-dependent slowdown for momentum-smoothed sign updates (Theorem 3.4).
- **General Nonconvex Convergence.** Under bounded-gradient stochastic nonconvex assumptions, we prove a general convergence guarantee that, with $\beta_2 = 1 - 1/T$, recovers the standard $O(1/\sqrt{T})$ rate up to the cost of variance normalization (Theorem 3.5).
- **Empirical Performance and Robustness.** On CIFAR-100 and GPT-style language modeling, MVN-Grad matches or improves on Adam(W), AdaBelief, and LaProp, while often improving robustness across the evaluated hyperparameter sweeps at the cost of one additional optimizer-state tensor.

Additional related work. Several works proposed Adam variants targeting stability and convergence (AMSGrad (Reddi et al., 2019), Yogi (Zaheer et al., 2018), RAdam (Liu et al., 2019)), practical training refinements (AdamW (Loshchilov et al., 2017), AdaFactor (Shazeer & Stern, 2018)), or stronger preconditioning (Sophia (Liu et al., 2023), AdaHessian (Yao et al., 2021)). These methods address complementary aspects of adaptive methods; we review them in Appendix A.

2 Proposed Algorithm

We consider stochastic optimization of a differentiable function F, observing $g_t = \nabla_x f(x_{t-1}; \xi_t)$ at iteration t. Updates are element-wise; for readability, we present the scalar form.

We introduce *Momentum on Variance-Normalized Gradients* (MVN-Grad), an Adam-style optimizer defined by two complementary choices: *(i)* variance-based normalization and *(ii)* applying

Algorithm 1 MVN-Grad (Momentum on Variance-Normalized Gradients)

Require: step size η, $\beta_1, \beta_2 \in [0, 1)$, $\varepsilon > 0$, $\varepsilon_s \geq 0$

Initialize $m_0 \leftarrow 0$, $s_0 \leftarrow 0$, $u_0 \leftarrow 0$, parameters x_0
for $t = 1, 2, \ldots$ **do**
  $g_t \leftarrow \nabla_x f(x_{t-1}; \xi_t)$
  $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1)\, g_t$
  $s_t \leftarrow \beta_2 s_{t-1} + (1 - \beta_2)\, (g_t - m_t)^2 + \varepsilon_s$
  $c_m \leftarrow 1 - \beta_1^t; \;\; c_v \leftarrow 1 - \beta_2^t$
  $z_t \leftarrow g_t / (\sqrt{\frac{s_t}{c_v}} + \varepsilon)$
  $u_t \leftarrow \beta_1 u_{t-1} + (1 - \beta_1)\, z_t$
  $x_t \leftarrow x_{t-1} - \eta\, u_t / c_m$
end for

momentum *after* normalization. Concretely, MVN-Grad replaces Adam's second-moment EMA normalizer with a variance estimator and applies momentum to normalized gradients rather than normalizing the momentum itself.

Algorithm 1 summarizes the update: m_t centers the variance proxy s_t, the bias-corrected s_t normalizes g_t, and u_t accumulates momentum on the resulting normalized gradients.

Implementation notes. Algorithm 1 reuses β_1 for both the mean estimator m_t and the momentum buffer u_t to match Adam's interface; decoupling them may help in non-stationary settings where variance estimation and momentum evolve on different timescales. Because ε_s is added at every iteration, the bias-corrected estimate satisfies $s_t/(1 - \beta_2^t) \geq \varepsilon_s/(1 - \beta_2)$, so ε_s acts as a tunable variance floor. Our theory sets $\varepsilon_s = 0$; in practice, it is tuned for the variance-normalized methods.

Taxonomy of Adaptive Architectures. Adaptive methods can be categorized along two orthogonal axes. The first axis is the choice of *normalizer*. Standard methods track the uncentered second moment $v_t \approx \mathbb{E}[g_t^2]$, whereas variance-based methods track the centered second moment $s_t \approx \mathrm{Var}(g_t)$. Let $\mathrm{EMA}_\beta(x_t)$ denote the exponential moving average of a sequence x_t with decay β. Using momentum m_t as the mean estimate, the normalizers are:

$$v_t = \mathrm{EMA}_{\beta_2}(g_t^2) \quad \text{vs} \quad s_t = \mathrm{EMA}_{\beta_2}\big((g_t - m_t)^2\big). \tag{1}$$

The second axis is the *ordering*. Let $r_t \in \{v_t, s_t\}$ denote the chosen normalizer. *Momentum-then-normalize* methods aggregate gradients before scaling, while *normalize-then-momentum* methods scale the gradient before aggregation. Ignoring bias corrections, the two update directions are:

$$\Delta_t^{\text{pre}} = \frac{\mathrm{EMA}_{\beta_1}(g_t)}{\sqrt{r_t} + \varepsilon} \quad \text{vs} \quad \Delta_t^{\text{post}} = \mathrm{EMA}_{\beta_1}\left(\frac{g_t}{\sqrt{r_t} + \varepsilon}\right). \tag{2}$$

In the first case (Δ_t^{pre}), the EMA is applied to the raw gradient g_t. In the second case (Δ_t^{post}), the EMA acts on the normalized gradient, serving as the final operation. These choices define a complete 2×2 design space. The *momentum-then-normalize* form (Δ_t^{pre}) includes Adam with second-moment scaling ($r_t = v_t$) and AdaBelief with variance scaling ($r_t = s_t$). The *normalize-then-momentum* form (Δ_t^{post}) includes LaProp (second-moment scaling) and MVN-Grad (variance normalization).

Why the two axes matter. MVN-Grad differs from Adam along two complementary design axes: normalizer choice and ordering, each addressing a distinct failure mode. We motivate both modifications below and formalize the resulting guarantees in Section 3.

(A) Choosing the ordering. Swapping normalization and momentum changes the *temporal dependency* of the update. Momentum-then-normalize methods (Adam, AdaBelief) divide the momentum buffer m_t by the current normalizer r_t. Since $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$ contains both past and current gradients, the update can be decomposed as $\Delta_t^{\text{pre}} = \beta_1 m_{t-1}/(\sqrt{r_t}+\varepsilon) + (1-\beta_1) g_t/(\sqrt{r_t}+\varepsilon)$. The first term divides a history-dependent quantity m_{t-1} by the present-time statistic r_t, creating a risky "cross-time" ratio. If the normalizer transiently dips while the momentum buffer still reflects a past spike, this contribution can be artificially amplified. In contrast, normalize-then-momentum forms the ratio $g_t/\sqrt{r_t}$ at the *same timestep*, ensuring the gradient is stabilized *before* entering the momentum buffer and eliminating this coupling; as shown by the toy example in Appendix Fig. 3a.

This behavior is not merely heuristic: the structural change yields two formal stability gains. First, Section 3.1 shows that applying momentum after normalization reduces the one-step conditional variance by eliminating stochastic interaction between carry-over momentum and a random denominator. Second, this ordering ensures isolated gradient spikes are clipped immediately, formalized in Section 3.2. Unlike Adam, where spikes are re-emitted over many steps, MVN-Grad scales outliers at the source.

(B) Choosing the normalizer. Switching from the second-moment normalizer v_t to the variance proxy s_t changes the *scale* of the update. The second moment mixes signal and noise and scales with $\mathbb{E}[g_t^2] \approx \mu^2 + \sigma^2$, whereas the variance s_t isolates the noise level σ^2. This distinction is especially pronounced in low-variance regimes ($\sigma \ll \mu$), which are illustrated by the toy example in Appendix Fig. 3b. In such settings, second-moment normalization effectively rescales the gradient by its own magnitude ($g_t/\sqrt{v_t} \approx g_t/|g_t|$), causing the method to collapse to a conservative, sign-based update. In contrast, variance normalization scales deviations around the mean, preserving informative gradient magnitudes up to a noise-dependent preconditioner. Section 3.3 illustrates this with two idealized limiting recursions.

3 Theoretical Guarantees

This section isolates MVN-Grad's two design mechanisms. Under a local symmetry and exact mean-tracking condition, we first show that normalize–then–momentum yields no larger one-step conditional update variance than momentum–then–normalize. We then characterize carry-over under an isolated-spike model, compare idealized high-signal normalization dynamics, and establish a general bounded-gradient nonconvex convergence guarantee. Before presenting our results, we fix notation. We analyze coordinate-wise and let g_t denote the stochastic gradient at step t for a fixed coordinate. All random variables are defined on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mathbb{P})$, where $\mathcal{F}_{t-1}$ captures the history up to time $t-1$. Let $\mu_t := \mathbb{E}[g_t \mid \mathcal{F}_{t-1}]$ denote the conditional mean. For clarity, we omit bias-correction factors of the form $1/(1-\beta^t)$, as they do not affect our conclusions. We denote the update directions for Adam and AdaBelief by Δ_t^{A} and Δ_t^{AB}, corresponding to the *momentum-then-normalize* structure Δ_t^{pre} in equation 2 with $r_t = v_t$ and $r_t = s_t$ from equation 1, respectively. Likewise, LaProp and MVN-Grad are denoted by Δ_t^{LP} and Δ_t^{MV}, corresponding to the *normalize-then-momentum* structure Δ_t^{post} with $r_t = v_t$ and $r_t = s_t$, respectively.

3.1 Conditional variance comparison

We quantify the stability gain of applying momentum *after* normalization while using the same variance proxy s_t. Specifically, we compare the one-step *conditional update variance* of AdaBelief and MVN-Grad, $\mathrm{Var}(\Delta_t^{\mathrm{AB}} \mid \mathcal{F}_{t-1})$ versus $\mathrm{Var}(\Delta_t^{\mathrm{MV}} \mid \mathcal{F}_{t-1})$. This conditional variance measures the unpredictability of the next update given the past, with smaller values indicating smoother dynamics. Under a standard symmetry assumption on centered gradient noise and an idealized mean-tracking condition on the EMA, the variance gap has a closed form and is always nonnegative. Throughout this subsection, we assume that the relevant conditional second moments are finite; in particular, $\mathbb{E}[g_t^2 \mid \mathcal{F}_{t-1}] < \infty$ almost surely.

Theorem 3.1. *(Proof in Appendix B.1) Assume that, conditional on $\mathcal{F}_{t-1}$, the innovation $\zeta_t := g_t - m_{t-1}$ is symmetric about zero. Define $\Delta\mathrm{Var}_t := \mathrm{Var}\big[\Delta_t^{AB} \mid \mathcal{F}_{t-1}\big] - \mathrm{Var}\big[\Delta_t^{MV} \mid \mathcal{F}_{t-1}\big]$. Then, we have the following $\Delta\mathrm{Var}_t = (2\beta_1 - \beta_1^2)\, m_{t-1}^2\, \mathrm{Var}(\frac{1}{\sqrt{s_t}+\varepsilon} \mid \mathcal{F}_{t-1}) \geq 0$.*

Theorem 3.1 isolates a concrete stability gap between the two orderings. Even with the same variance proxy, AdaBelief multiplies carried momentum by the current random normalizer, whereas MVN-Grad's carry-over is $\mathcal{F}_{t-1}$-measurable. The gap is nonnegative and is strictly positive whenever $\beta_1 > 0$, $m_{t-1} \neq 0$, and the normalizer is conditionally non-degenerate. Although the theorem assumes local symmetry, the MNIST and CIFAR-100 diagnostics exhibit the same positive gap along actual training trajectories.

Appendix Fig. 4 reports a frozen-checkpoint Monte Carlo diagnostic on MNIST. The estimated variance gap remains positive on average across checkpoints, providing empirical support for the normalize–then–momentum mechanism. A corresponding CIFAR-100 diagnostic is reported in Fig. 1.

Proposition 3.2. *(Proof in Appendix B.2) Adopt the setup of Theorem 3.1, but replace the symmetry assumption with the following: conditional on $\mathcal{F}_{t-1}$, the stochastic gradient admits $g_t = \mu_t + \xi_t$ where ξ_t is symmetric about zero, $\mathbb{E}[\xi_t^2 \mid \mathcal{F}_{t-1}] \le \sigma_\xi^2$, and the EMA tracking error satisfies $|m_{t-1} - \mu_t| \le \delta$. Define $Y_t := (\sqrt{s_t} + \varepsilon)^{-1}$. Then: $\Delta\mathrm{Var}_t \ge (2\beta_1 - \beta_1^2)\, m_{t-1}^2 \,\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] - 2\beta_1(1-\beta_1)\,|m_{t-1}|\,\big[\delta\,\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] + \sigma_\xi/\varepsilon^2\big]$. In particular, the gap remains non-negative whenever $|m_{t-1}|$ is sufficiently large relative to δ and σ_ξ/ε^2, i.e., $(2-\beta_1)|m_{t-1}|\,\mathrm{Var}[Y_t] \ge 2(1-\beta_1)[\delta\,\mathrm{Var}[Y_t] + \sigma_\xi/\varepsilon^2]$.*

This condition is sufficient rather than necessary, and is most informative when the carry-over momentum $|m_{t-1}|$ is large relative to the local mean-tracking error and noise–normalizer term. If $\mathrm{Var}(Y_t \mid \mathcal{F}_{t-1}) = 0$, the normalizer is conditionally deterministic and the exact variance gap in Theorem 3.1 is zero.

3.2 Spike response and momentum carry-over

To make the ordering effect in Section 3.1 concrete, we study a spike-response model that isolates whether a large gradient is normalized before entering the momentum buffer or stored first and attenuated only later. We begin with a single spike, where the mechanism admits a clean closed-form analysis, and then give an extension to later spikes that dominate the current mean estimate. The purpose of this section is not to claim universal robustness to arbitrary heavy-tailed sequences, but to isolate a concrete carry-over mechanism induced by the ordering of momentum and normalization.

Theorem 3.3. *(Proof in Appendix B.3) Consider $g_0 = Mu$ with $M \ge 1$ and $u \ne 0$, and $g_t = u$ for all $t \ge 1$. Assume $\beta_1 \in (0,1)$, $\beta_2 \in [0,1)$, $\varepsilon > 0$, $m_{-1} = 0$, $u_{-1} = 0$ for the post-normalization momentum buffers, and $v_{-1} = s_{-1} = \bar{d} > 0$. For LaProp and MVN-Grad, the update magnitudes are uniformly bounded: $|\Delta_t^{\mathrm{LP}}| \le C$, $|\Delta_t^{\mathrm{MV}}| \le C$, $\forall\, t \ge 0$, where C depends only on $(\beta_1, \beta_2, \bar{d}, u, \varepsilon)$ and not on M. In contrast, for Adam if $t^\star := \min\Big\{t \ge 1 :\ (1-\beta_2)\beta_2^t M^2 \le 1\Big\}$, then there exists $c > 0$, depending only on $(\beta_1, \beta_2, \bar{d}, u, \varepsilon)$, such that $|\Delta_{t^\star}^{\mathrm{A}}| \ge c\,(1-\beta_1)\beta_1^{t^\star} M$.*

Theorem 3.3 shows a contrast in how a spike propagates through the optimizer state. For Adam, the spike enters the raw momentum m_0 and decays as $\beta_1^t M$, while v_t forgets it as $\beta_2^t M^2$. At time $t^\star$ when the spike contribution in v_t returns to baseline, Adam retains a carry-over of order $(1-\beta_1)\beta_1^{t^\star} M$ in the numerator. In contrast, LaProp and MVN-Grad normalize the spike at $t = 0$ before it enters the momentum buffer, so the resulting updates cannot reintroduce a factor growing with M.

Appendix Fig. 5 corroborates Theorem 3.3 in two stress-test regimes: LaProp and MVN-Grad remain bounded as M grows, whereas Adam exhibits spike-dependent growth when the denominator forgets the spike faster than the numerator. AdaBelief also exhibits amplification in one of the evaluated regimes because of its variance–momentum coupling.

Remark 3.1. When $\beta_2 > \beta_1^2$, Adam's update is uniformly bounded: $|\Delta_t^{\mathrm{A}}| \le \frac{1-\beta_1}{\sqrt{1-\beta_2}} \cdot \frac{1}{\sqrt{1-\beta_1^2/\beta_2}}$ for all $t \ge 0$. In this regime, the forgetting time $t^\star$ grows logarithmically with M, so $\beta_1^{t^\star}$ decays and the lower bound in Theorem 3.3 becomes loose. Outside this regime (e.g., $\beta_1 \approx 1$ with smaller β_2), the momentum remnant can yield spike-dependent growth over practical M, as shown in Appendix Fig. 5.

Corollary 3.3.1. *(Proof in Appendix B.4) For LaProp, the normalized input satisfies $|z_t| \le 1/\sqrt{1-\beta_2}$ at every iteration. For MVN-Grad with $\beta_1 > 0$, whenever $|g_t| \ge 2|m_{t-1}|$, we have $|z_t| \le 2/(\beta_1\sqrt{1-\beta_2})$. Thus, each such spike enters the momentum buffer with a contribution bounded independently of its magnitude.*

3.3 Second moment vs. variance normalization

We now consider a complementary idealized low-variance regime in which gradients align closely with the signal. Here, the second moment mixes signal and noise ($\mathbb{E}[g_t^2] = (\mathbb{E}[g_t])^2 + \mathrm{Var}(g_t)$), whereas variance isolates the noise. When $\mathrm{Var}(g_t) \ll (\mathbb{E}[g_t])^2$, second-moment normalization can become sign-like and largely discard gradient-magnitude information. Conversely, when the variance estimate is stable, variance normalization preserves magnitude information up to a noise-dependent preconditioner.

To formalize this, we adopt a standard local approximation for exponential moving averages. Assuming the gradient distribution varies slowly over the effective window $(1-\beta_2)^{-1}$, we decompose the stochastic gradient into a deterministic signal and a slowly varying noise term. Under this locally stationary approximation, bias-corrected EMAs track the gradient's second moment and variance over the relevant window. This requires only local moment stability and concentration of the EMA estimates over this window, rather than global stationarity.

Assumption 3.1. *At iteration t, the stochastic gradient admits the decomposition $g_t = \nabla F(x_{t-1}) + \xi_t$, where $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0$. For each coordinate i, suppose that over the effective EMA window the conditional noise variance is approximately constant,$\mathrm{Var}(\xi_{t,i} \mid \mathcal{F}_{t-1}) \approx \sigma_i^2$, and that the bias-corrected EMA estimates are locally concentrated around the corresponding moments: $v_{t,i} \approx \nabla_i F(x_{t-1})^2 + \sigma_i^2$, and $s_{t,i} \approx \sigma_i^2$.*

Under Assumption 3.1, both LaProp and MVN-Grad admit the coordinate-wise recursion $x_t = x_{t-1} - \eta u_t$, and $u_t = \beta_1 u_{t-1} + (1-\beta_1) z_t$, but they differ in the normalized input: $z_{t,i}^{\mathrm{LP}} = \frac{g_{t,i}}{\sqrt{v_{t,i}}+\varepsilon_{\mathrm{den}}}$, and $z_{t,i}^{\mathrm{MV}} = \frac{g_{t,i}}{\sqrt{s_{t,i}}+\varepsilon_{\mathrm{den}}}$. Thus, the comparison reduces to characterizing these directions in the low-variance regime. To isolate the effect of the normalizer and suppress stochastic fluctuations, we formalize this regime as follows.

Assumption 3.2. *Along the trajectory, $g_{t,i} \approx \nabla_i F(x_{t-1})$, and $\sigma_i^2 \ll \nabla_i F(x_{t-1})^2$. Moreover, on coordinates for which we invoke the sign approximation, $\varepsilon_{\mathrm{den}} \ll |\nabla_i F(x_{t-1})|$.*

Under Assumptions 3.1–3.2, the two normalization schemes admit distinct limiting behaviors. We analyze them in turn. For LaProp, which uses second-moment normalization, $v_{t,i} \approx \nabla_i F(x_{t-1})^2 \approx g_{t,i}^2$. Hence $z_{t,i}^{\mathrm{LP}} = \frac{g_{t,i}}{\sqrt{v_{t,i}}+\varepsilon_{\mathrm{den}}} \approx \frac{g_{t,i}}{|g_{t,i}|+\varepsilon_{\mathrm{den}}} = \frac{\mathrm{sign}(g_{t,i})}{1+\varepsilon_{\mathrm{den}}/|g_{t,i}|}$. Therefore, whenever $|g_{t,i}| \gg \varepsilon_{\mathrm{den}}$, $z_{t,i}^{\mathrm{LP}} \approx \mathrm{sign}(g_{t,i}) \approx \mathrm{sign}(\nabla_i F(x_{t-1}))$. Thus, in the low-variance high-signal regime, LaProp behaves approximately like a momentum-smoothed sign method and largely discards gradient-magnitude information. In contrast, MVN-Grad uses variance normalization. Under Assumption 3.1, $s_{t,i} \approx \sigma_i^2$, and therefore $z_{t,i}^{\mathrm{MV}} = \frac{g_{t,i}}{\sqrt{s_{t,i}}+\varepsilon_{\mathrm{den}}} \approx \frac{\nabla_i F(x_{t-1})}{\sigma_i+\varepsilon_{\mathrm{den}}}$. Consequently, a locally stable variance normalizer preserves gradient magnitudes up to a fixed coordinate-wise preconditioner rather than collapsing to a sign direction.

The relations above motivate two limiting recursions: a fixed normalizer for MVN-Grad and a sign normalizer for LaProp. The following theorem analyzes these limiting recursions exactly. Thus, the approximations in Assumptions 3.1–3.2 motivate the reduction, while the theorem itself makes exact statements about the resulting surrogate dynamics. Here σ should be interpreted as a locally stabilized normalization scale over the finite regime under consideration, rather than as a fixed global noise floor persisting arbitrarily close to stationarity.

Theorem 3.4. *(Proof in Appendix B.5) Let F be L-smooth and bounded below on $\mathbb{R}^d$. Consider the following exact limiting recursions, motivated by Assumptions 3.1–3.2. Assume an oracle $g_t = \nabla F(x_{t-1}) + \xi_t$ with $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0$ and $|\xi_{t,i}| \le \delta |\nabla_i F(x_{t-1})|$ almost surely, where $\delta \in (0,1)$. Let $\beta_1 \in [0,1)$ and initialize the momentum at zero.*

(i) (MVN-Grad, upper bound.) Fix a normalization scale $\sigma > 0$ and consider the stabilized limiting recursion with $z_t = g_t/(\sigma + \varepsilon_{\mathrm{den}})$. Define $\alpha := \frac{\eta(1-\beta_1)}{\sigma+\varepsilon_{\mathrm{den}}}$ and assume $\alpha \le (1-\beta_1^2)/L$. Then it suffices to take $T = \mathcal{O}(1/\varepsilon^2)$ iterations to ensure $\min_{0\le t\le T} \mathbb{E}\|\nabla F(x_t)\|_2 \le \varepsilon$.

(ii) (LaProp, lower bound.) For every horizon T, there exists an L-smooth function F (satisfying the same assumptions and oracle model above) and an initialization $x_0 = x_0(T)$ such that, for the corresponding limiting sign recursion $z_t^{\mathrm{LP}} = \mathrm{sign}(g_t)$, the following holds. For LaProp with step size $\eta_t = \frac{c}{L\sqrt{t+1}}$, $\min_{0\le t\le T} \|\nabla F(x_t)\|_2 = \frac{c(1-\beta_1^T)\sqrt{d}}{2\sqrt{T}}$ a.s. Consequently, whenever $1-\beta_1^T \ge \frac{1}{2}$, achieving $\min_{0\le t\le T} \|\nabla F(x_t)\|_2 \le \varepsilon$ requires $T = \Omega\big(\frac{d}{\varepsilon^2}\big)$ iterations.

Theorem 3.4 gives a dimension-dependent finite-horizon separation between the two idealized normalization dynamics: the fixed-normalizer recursion preserves gradient magnitude, whereas the sign recursion leaves a residual of order $\sqrt{d/T}$. This is not a standard oracle-complexity separation because the initialization in part (ii) depends on T and satisfies $F(x_0) - F^\star = \Theta(c^2 dT/L)$. Extending the result to the full finite-time EMA recursions would additionally require controlling the deviations of s_t and v_t from their stabilized limits.

Table 1: CIFAR-100 / ResNet-18: accuracy (mean ± std over 3 seeds) for the selected configuration.

Batch size 128		
Optimizer	**Test Acc.**	**Train Acc.**
Adam	77.82 ± 0.18	75.65 ± 0.71
LaProp	77.72 ± 0.20	75.59 ± 0.99
AdaBelief	79.93 ± 0.15	81.22 ± 0.88
MVN-Grad	$\mathbf{79.94 \pm 0.17}$	$\mathbf{81.26 \pm 0.94}$

Batch size 1024		
Optimizer	**Test Acc.**	**Train Acc.**
Adam	78.26 ± 0.14	83.15 ± 1.28
LaProp	78.40 ± 0.14	83.05 ± 1.34
AdaBelief	79.34 ± 0.27	85.89 ± 1.17
MVN-Grad	$\mathbf{79.63 \pm 0.12}$	$\mathbf{85.89 \pm 1.19}$

3.4 General convergence guarantee

Unlike prior results so far, we next establish MVN-Grad's general nonconvex convergence under standard stochastic conditions, without requiring high-SNR or stabilized-variance regimes.

Theorem 3.5. *(Proof in Appendix B.6) Let $F : \mathbb{R}^d \to \mathbb{R}$ be L-smooth with $F_\star := \inf F > -\infty$. Let $g_t = \nabla F(x_{t-1}) + \xi_t$ with $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0$, $\|g_t\|_\infty \le G$ a.s., and $\mathbb{E}[\xi_{t,i}^2 \mid \mathcal{F}_{t-1}] \le \sigma_\xi^2$. Consider the uncorrected MVN-Grad recursion obtained from Algorithm 1 by omitting bias-correction factors, with $\varepsilon_s = 0$ and zero-initialized optimizer states $m_0 = s_0 = u_0 = 0$, constant learning rate $\eta > 0$, momentum $\beta_1 \in (0, 1)$, EMA parameter $\beta_2 \in [0, 1)$, and $\varepsilon > 0$. Also, assume that $\varepsilon \le G$ and that the stepsize satisfies $\eta \le c\frac{(1-\beta_1)^2\varepsilon}{L\beta_1^2}$ for a constant $c > 0$. Then*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\Big[\|\nabla F(x_t)\|^2\Big] \le \frac{C_1 G \Delta_0}{\eta T} + C_1 L \eta \frac{dG^3}{\varepsilon^2} + \frac{C_2 d(1-\beta_2) G^3 \sigma_\xi^3}{\varepsilon^4}, \tag{3}$$

where $\Delta_0 := F(x_0) - F_\star$, and $C_1, C_2 > 0$ depend only on β_1.

The final term in equation 3 is a non-vanishing term in the bound caused by the correlation between g_t and its variance-based normalizer. Choosing $\beta_2 = 1 - 1/T$ reduces this term to $O(dG^3\sigma_\xi^3/(T\varepsilon^4))$ and, after optimizing η, recovers the standard $O(1/\sqrt{T})$ dependence, with the ε^{-1} factor reflecting worst-case control when the variance proxy becomes small. The proof uses a virtual-iterate reduction and covariance decomposition; details are in Appendix B.6. This general guarantee complements the idealized high-signal result above, where a stabilized variance normalizer preserves gradient magnitude.

4 Experiments

We compare MVN-Grad with Adam, AdaBelief, and LaProp on CIFAR-100/ResNet-18 and GPT-style language modeling. Non-optimizer hyperparameters are fixed, and optimizer hyperparameters are tuned using a coarse-to-focused sweep. Tables report selected-configuration mean ± std over seeds. Boxplots aggregate all sweep runs: better-positioned boxes indicate stronger performance across the tuning grid, while narrower boxes and shorter tails indicate lower hyperparameter sensitivity.

4.1 CIFAR-100 with ResNet-18

We train ResNet-18 on CIFAR-100 for 200 epochs with cosine learning-rate decay, warmup, weight decay, RandAugment, MixUp, and label smoothing. We consider batch sizes 128 and 1024 to probe different gradient-noise regimes. We use shared grids over learning rate and (β_1, β_2); variance-normalized methods additionally tune their method-specific parameter ε_s. We select the best configuration for each optimizer using a single selection seed and evaluate it over three seeds.

Table 1 reports final accuracies; Figs. 2a, 2d, 2b, and 2e show robustness via boxplots over all sweep runs. At batch size 128, MVN-Grad matches AdaBelief and improves over Adam and LaProp by 2.1–2.2% test accuracy, with comparable sweep medians (77.72% vs. 77.94%) and shorter left tails. At batch size 1024, MVN-Grad achieves the best mean test accuracy (79.63%, +0.29% over AdaBelief) and the strongest sweep robustness: highest median test accuracy (78.33% vs. 76.41% for AdaBelief, $\approx$ 74% for Adam/LaProp) and highest median train accuracy (81.65% vs. 73.87% for AdaBelief, $\approx$ 65% for Adam/LaProp). Fig. 1 further supports the mechanism of Section 3.1 on CIFAR-100: the variance gap remains positive across checkpoints.

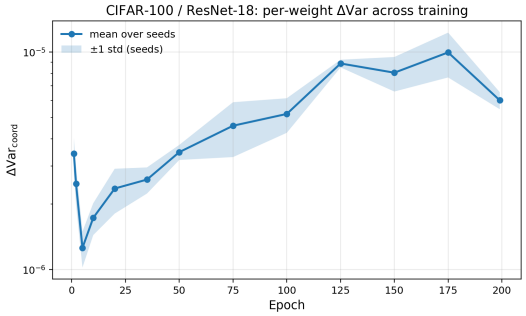
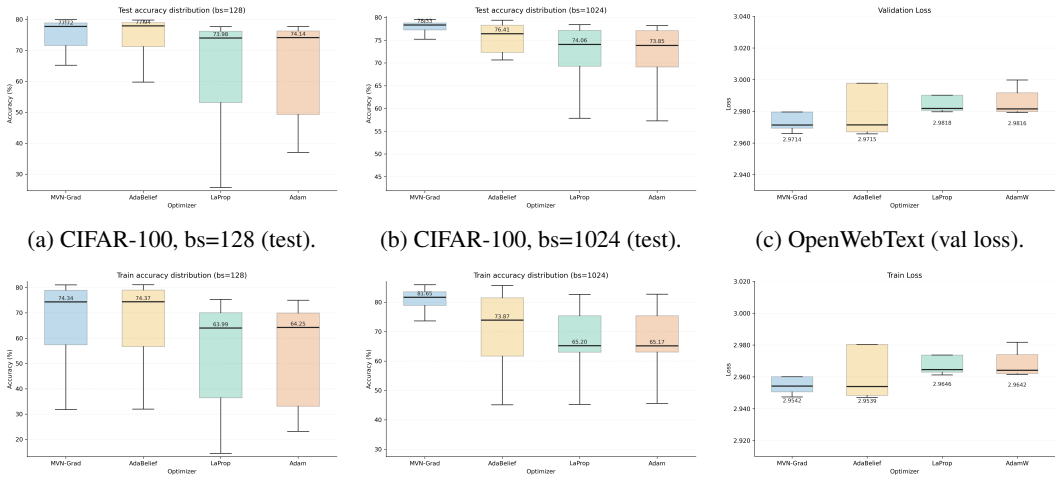


Figure 1: Update-variance gap on CIFAR-100/ResNet-18. The gap remains positive throughout training, supporting the mechanism of Theorem 3.1.



(a) CIFAR-100, bs=128 (test). (b) CIFAR-100, bs=1024 (test). (c) OpenWebText (val loss).

(d) CIFAR-100, bs=128 (train). (e) CIFAR-100, bs=1024 (train). (f) OpenWebText (train loss).

Figure 2: Hyperparameter robustness across sweep runs: validation (top) and training (bottom).

4.2 Language Modeling

We evaluate GPT-style language models with 30M parameters on WikiText-103, 124M parameters on OpenWebText, and 350M parameters on FineWeb. We report cross-entropy loss, with means and standard deviations computed from the individual seed-level losses. The corresponding training losses are reported in Table 10 in the appendix.

WikiText-103. We train a 30M-parameter GPT-2–style Transformer with sequence length 512 on WikiText-103. All results are reported as mean $\pm$ standard deviation over 4 seeds. We use a two-stage tuning protocol with shared learning-rate and (β_1, β_2) grids across optimizers; variance-normalized methods additionally tune ε_s. Full grids are in Appendix C.6. The WikiText-103 experiments use *coupled* ℓ_2 regularization, in which the term λx_{t-1} is added to the stochastic gradient prior to moment estimation. Under coupled decay, the regularization signal is filtered through the optimizer's internal states, and its effect depends on update ordering—specifically, whether momentum is applied before normalization or after normalization. This interaction can increase seed sensitivity. For the larger-scale OpenWebText experiments below, we instead use *decoupled* weight decay to enable a cleaner comparison. As shown in Table 2, MVN-Grad achieves the lowest mean validation loss, reducing it by 0.030 compared to Adam and 0.015 compared to AdaBelief. Across seeds, validation loss exhibits the largest standard deviation for LaProp, followed by MVN-Grad.

OpenWebText. We train a GPT-2 124M model on OpenWebText with context length 1024, using micro-batches of size 12 with 40 gradient-accumulation steps (effective batch size 480), bf16 precision, gradient clipping at 1, 2000 warmup steps, and 120k training iterations. We disable dropout and use AdamW-style *decoupled* weight decay with coefficient 0.1. Based on preliminary runs at

Table 2: Language-modeling validation loss (mean ± standard deviation) for the selected configurations. Method names denote optimizer families; regularization and clipping follow the dataset-specific protocols described in the text. Training losses and detailed hyperparameter results are reported in the appendix.

Optimizer	WikiText-103 30M	OpenWebText 124M	FineWeb 350M
Adam / AdamW	4.167 ± 0.018	2.990 ± 0.010	3.2360 ± 0.0065
LaProp / LaPropW	4.202 ± 0.076	2.990 ± 0.010	3.2361 ± 0.0060
AdaBelief / AdaBeliefW	4.152 ± 0.018	2.974 ± 0.008	3.2350 ± 0.0077
MVN-Grad / MVN-GradW	$\mathbf{4.137} \pm 0.040$	$\mathbf{2.973} \pm 0.007$	$\mathbf{3.2344} \pm 0.0089$

alternative learning rates, we identify $\eta = 10^{-4}$ as a stable common learning-rate regime and then evaluate the same (β_1, β_2) grid for each optimizer; variance-normalized methods use $\varepsilon_s = 0$. We select by final validation loss and re-run with three seeds; see Appendix C.6. Table 2 reports final losses for the selected configurations, while Figs. 2c and 2f visualize robustness via boxplots over all evaluated hyperparameters. MVN-Grad achieves the best mean losses, slightly improving over AdaBeliefW and more clearly outperforming AdamW and LaPropW. Across the full sweep, MVN-GradW attains the lowest median validation loss (2.9714), matching AdaBeliefW (2.9715) and outperforming AdamW and LaPropW (both $\approx$ 2.9816–2.9818), with similarly competitive train-loss medians. Moreover, normalize–then–momentum methods show narrower interquartile ranges than momentum–then–normalize methods, indicating less sensitivity to (β_1, β_2) at a fixed learning rate.

FineWeb (350M). We extend the evaluation to a 350M-parameter GPT-style model on FineWeb (Penedo et al., 2024), trained for 4,000 iterations. All four optimizers use the same architecture, training-data setting, momentum parameters, and final-update row-clipping rule. Preliminary experiments identified $(\beta_1, \beta_2) = (0.9, 0.95)$ as a stable common momentum setting across all four optimizers. We therefore fix this pair for the 350M comparison and sweep only the learning rate, keeping the larger-scale evaluation compute-matched across methods. The full seed-1 learning-rate comparison is reported in Appendix Table 9. At this configuration, MVN-Grad obtains a four-seed validation loss of 3.2344 ± 0.0089, compared with 3.2350 ± 0.0077 for AdaBelief, 3.2360 ± 0.0065 for AdamW, and 3.2361 ± 0.0060 for LaProp. These closely matched results extend MVN-Grad's competitive performance to the larger model scale.

Memory and runtime overhead. MVN-GradW maintains three full-size state tensors per parameter (m_t, s_t, u_t), whereas AdamW, LaPropW, and AdaBeliefW maintain two. Counting parameters, gradients, and optimizer states, this increases parameter-sized storage from approximately $4\times$ to $5\times$ model size. In wall-clock time, MVN-GradW remains comparable to the baselines on OpenWebText: AdamW takes 46.2k s, MVN-GradW 46.8k s, LaPropW 47.4k s, and AdaBeliefW 48.2k s. MVN-Grad is therefore most attractive when optimizer-state memory is affordable and reduced sensitivity to tuning or update fluctuations is valuable. On FineWeb with the 350M model, mean selected-configuration runtimes across four seeds ranged from 8.95k s to 8.99k s across optimizers (about 2.5 hours per run), with MVN-GradW remaining within 0.5% of AdamW.

5 Conclusion

We proposed *MVN-Grad*, an Adam-style optimizer that combines variance-based normalization with normalize–then–momentum ordering. Our analysis separated the roles of these two choices: the ordering reduces conditional update variance and bounds spike sensitivity, while variance normalization avoids sign collapse and the resulting dimension-dependent slowdowns in the small-variance (high-signal) regime. On CIFAR-100 and GPT-style language modeling benchmarks, MVN-Grad matched or improved over Adam, AdaBelief, and LaProp, with improved sweep robustness and comparable wall-clock cost. Our mechanism-isolating theorems rely on local symmetry and high-SNR conditions that clarify structural advantages but do not cover all training regimes; the general convergence rate carries an ε^{-1} cost from worst-case variance normalization. Extending the analysis to heavy-tailed noise, larger-scale pretraining, and sharper stabilized-regime rates are directions for future work.

AI USE STATEMENT

We used generative AI tools to assist with improving the clarity, grammar, and presentation of the manuscript, and as an auxiliary tool for checking the consistency and correctness of mathematical derivations and arguments. Generative AI tools were also used to assist with writing and debugging portions of the experimental code.

All AI-assisted text, mathematical suggestions, and code were critically reviewed by the authors. Theoretical claims and proofs were independently verified by the authors, and AI-assisted experimental code was inspected, tested, and validated through the resulting experiments. Generative AI was not used to generate or fabricate experimental results or data, nor to make final decisions regarding the scientific claims of the paper. We take full responsibility for the final content of this work, including all text, proofs, code, claims, and experimental results.

ETHICS STATEMENT

This work studies optimization algorithms and does not involve human subjects, private or sensitive personal data, or deployment in safety-critical settings. Our experiments use standard publicly available machine-learning benchmarks and datasets. We are not aware of any specific ethical concerns beyond the general environmental and societal considerations associated with computational machine-learning research.

REPRODUCIBILITY STATEMENT

We provide the complete MVN-Grad algorithm in Algorithm 1 and state all assumptions used in the theoretical analysis in Section 3. Complete proofs of the theoretical results are provided in Appendix B. Experimental setups, compute resources, hyperparameter-search procedures, and selected hyperparameters for MNIST, CIFAR-100, WikiText-103, OpenWebText, and FineWeb are reported in Appendix C. The main text and appendix also report seed-level aggregation procedures and the hyperparameter grids used for the comparisons, with the goal of enabling reproduction of both the theoretical and empirical results.

REFERENCES

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

Lukas Balles and Philipp Hennig. Dissecting adam: The sign, magnitude and variance of stochastic gradients. In *International Conference on Machine Learning*, pp. 404–413. PMLR, 2018.

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.

Jinghui Chen, Dongruo Zhou, Yiqi Tang, Ziyan Yang, Yuan Cao, and Quanquan Gu. Closing the generalization gap of adaptive gradient methods in training deep neural networks. *arXiv preprint arXiv:1806.06763*, 2018a.

Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. *arXiv preprint arXiv:1808.02941*, 2018b.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, Yang Zhang, and Jonathan M Cohen. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks. *arXiv preprint arXiv:1905.11286*, 2019.

Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

Vineet Gupta, Tomer Koren, and Yoram Singer. Shampoo: Preconditioned stochastic tensor optimization. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1842–1850. PMLR, 7 2018. URL https://proceedings.mlr.press/v80/gupta18a.html.

Byeongho Heo, Sanghyuk Chun, Seong Joon Oh, Dongyoon Han, Sangdoo Yun, Gyuwan Kim, Youngjung Uh, and Jung-Woo Ha. Adamp: Slowing down the slowdown for momentum optimizers on scale-invariant weights. *arXiv preprint arXiv:2006.08217*, 2020.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

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

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Hong Liu, Zhiyuan Li, David Hall, Percy Liang, and Tengyu Ma. Sophia: A scalable stochastic second-order optimizer for language model pre-training. *arXiv preprint arXiv:2305.14342*, 2023.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.

Liangchen Luo, Yuanhao Xiong, Yan Liu, and Xu Sun. Adaptive gradient methods with dynamic bound of learning rate. *arXiv preprint arXiv:1902.09843*, 2019.

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

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

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

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30, 2017.

Zhewei Yao, Amir Gholami, Sheng Shen, Mustafa Mustafa, Kurt Keutzer, and Michael Mahoney. Adahessian: An adaptive second order optimizer for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 10665–10673, 2021.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

Wei Yuan and Kai-Xin Gao. Eadam optimizer: How ε impact adam. *arXiv preprint arXiv:2011.02150*, 2020.

Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

Zhiming Zhou, Qingru Zhang, Guansong Lu, Hongwei Wang, Weinan Zhang, and Yong Yu. Adashift: Decorrelation and convergence of adaptive learning rate methods. *arXiv preprint arXiv:1810.00143*, 2018.

Juntang Zhuang, Tommy Tang, Yifan Ding, Sekhar C Tatikonda, Nicha Dvornek, Xenophon Papademetris, and James Duncan. Adabelief optimizer: Adapting stepsizes by the belief in observed gradients. *Advances in neural information processing systems*, 33:18795–18806, 2020.

Juntang Zhuang, Yifan Ding, Tommy Tang, Nicha Dvornek, Sekhar C Tatikonda, and James Duncan. Momentum centering and asynchronous update for adaptive gradient methods. *Advances in Neural Information Processing Systems*, 34:28249–28260, 2021.

Liu Ziyin, Zhikang T Wang, and Masahito Ueda. Laprop: Separating momentum and adaptivity in adam. *arXiv preprint arXiv:2002.04839*, 2020.

A Extended related work: Adam-style optimizers

A taxonomy of Adam variants. Most Adam-style optimizers modify (i) the adaptivity statistic and its update rule, (ii) the coupling between numerator and denominator (including update ordering or temporal decorrelation), (iii) step-size control (e.g., bounds or variance rectification), (iv) practical regularization details (e.g., weight decay), and/or (v) systems constraints such as memory footprint or large-batch training.

Convergence and stability fixes. A key theoretical line identifies settings where Adam can fail and proposes convergent variants such as AMSGrad (Reddi et al., 2019). Related stability-oriented modifications include controlling the growth of second-moment accumulators (Yogi (Zaheer et al., 2018)) and rectifying early-stage variance of the adaptive learning rate (RAdam (Liu et al., 2019)).

Decoupling momentum and adaptivity. Several methods aim to reduce harmful interactions between historical momentum and stochastic normalization. LaProp (Ziyin et al., 2020) applies momentum after normalization to separate these effects, which is closely related in spirit to our focus on ordering. AdaShift (Zhou et al., 2018) uses temporally shifted statistics to decorrelate numerator and denominator.

Alternative scaling rules. Beyond second-moment normalization, AdaBelief (Zhuang et al., 2020) adapts step sizes using deviations from the predicted gradient. This line targets scaling behavior and can be viewed as variance-sensitive normalization.

Practical refinements and generalization-motivated variants. AdamW (Loshchilov et al., 2017) decouples weight decay from the gradient step. Memory and implementation constraints motivated AdaFactor (Shazeer & Stern, 2018) and SM3 (Anil et al., 2019). Other variants aim to mitigate the adaptivity–generalization gap or extreme effective learning rates via bounded/partial adaptivity, including AdaBound (Luo et al., 2019) and Padam (Chen et al., 2018a). Further modifications address scale-invariant weights (AdamP (Heo et al., 2020)).

Large-batch and large-scale training. Layer-wise adaptations such as LAMB (You et al., 2019) and methods like NovoGrad (Ginsburg et al., 2019) target stability and efficiency in large-batch regimes.

Interpretations of Adam and sign-like behavior. Several works interpret Adam as combining sign-like directions with variance-dependent magnitudes (Balles & Hennig, 2018), and analyze generalization differences between adaptive methods and SGD (Wilson et al., 2017). These perspectives connect directly to the sign-collapse phenomenon discussed in Section 3.3.

Beyond diagonal second-moment preconditioning. Recent optimizers incorporate curvature information using inexpensive diagonal Hessian estimates and clipping, including Sophia (Liu et al., 2023) and related diagonal-curvature methods such as AdaHessian (Yao et al., 2021). We include these for context, though they go beyond strictly Adam-style diagonal second-moment preconditioning. Related preconditioning methods include Shampoo (Gupta et al., 2018) and Muon (Jordan et al.).

Summary table. For quick reference, Table 3 summarizes representative Adam-style variants, highlighting the core modification each method makes and the primary motivation behind it.

Method(s)	Key change	Main purpose
AMSGrad (Reddi et al., 2019)	Monotone denominator via max-correction	Convergence / avoid pathological steps
Yogi (Zaheer et al., 2018), RAdam (Liu et al., 2019)	Stabilize/rectify second-moment statistics	Reduce instability (esp. early / heavy variance)
AdaShift (Zhou et al., 2018)	Time-shifted statistics	Decorrelate numerator and denominator
LaProp (Ziyin et al., 2020)	Momentum *after* normalization	Reduce coupling of stale momentum and noisy scaling
AdaBelief (Zhuang et al., 2020)	Deviation/variance-sensitive scaling	Improve adaptivity when noise is small
AdamW (Loshchilov et al., 2017)	Decoupled weight decay	Correct regularization under adaptivity
AdaFactor (Shazeer & Stern, 2018), SM3 (Anil et al., 2019)	Memory-efficient accumulators	Scale to large models / reduced memory
AdaBound (Luo et al., 2019), Padam (Chen et al., 2018a)	Bounded/partial adaptivity	Control extreme per-coordinate LRs; SGD-like late stage
AdamP (Heo et al., 2020)	Projection for scale-invariant parameters	Improve behavior with norm/scale invariances
LAMB (You et al., 2019), NovoGrad (Ginsburg et al., 2019)	Layer-wise scaling / trust ratio	Large-batch stability and efficiency
Sophia (Liu et al., 2023), AdaHessian (Yao et al., 2021)	Diagonal curvature information	Curvature-aware scaling beyond second moments
Beyond Adam-style diagonal preconditioning (context): Shampoo (Gupta et al., 2018), Muon (Jordan et al.).		

Table 3: Representative variants around Adam (not exhaustive).

B THEORY

B.1 PROOF OF THEOREM 3.1

Proof. Fix an iteration t and define

$$Y_t := \frac{1}{\sqrt{s_t}+\varepsilon}, \qquad X_t := \frac{g_t}{\sqrt{s_t}+\varepsilon} = g_t\, Y_t.$$

We can write the updates as

$$\Delta_t^{AB} \;=\; \beta_1 m_{t-1} Y_t + (1-\beta_1) X_t,$$

and, since MVN-Grad applies momentum *after* normalization,

$$\Delta_t^{MV} \;=\; \beta_1 u_{t-1} + (1-\beta_1) X_t,$$

where u_{t-1} is $\mathcal{F}_{t-1}$-measurable.

Conditional variance gap identity. Because m_{t-1} and u_{t-1} are $\mathcal{F}_{t-1}$-measurable, we have

$$\begin{aligned}\mathrm{Var}\big[\Delta_t^{AB} \mid \mathcal{F}_{t-1}\big] &= \mathrm{Var}[\beta_1 m_{t-1} Y_t + (1-\beta_1) X_t \mid \mathcal{F}_{t-1}] \\ &= \beta_1^2 m_{t-1}^2 \mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] + (1-\beta_1)^2 \mathrm{Var}[X_t \mid \mathcal{F}_{t-1}] \\ &\quad + 2\beta_1(1-\beta_1) m_{t-1} \mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}]\,,\end{aligned}$$

and

$$\mathrm{Var}\big[\Delta_t^{MV} \mid \mathcal{F}_{t-1}\big] = \mathrm{Var}[\beta_1 u_{t-1} + (1-\beta_1) X_t \mid \mathcal{F}_{t-1}] = (1-\beta_1)^2 \mathrm{Var}[X_t \mid \mathcal{F}_{t-1}]\,.$$

Subtracting yields

$$\Delta\mathrm{Var}_t = \beta_1^2 m_{t-1}^2 \mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] + 2\beta_1(1-\beta_1) m_{t-1} \mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}]\,. \tag{4}$$

Evaluate the covariance term via symmetry. Let $\zeta := g_t - m_{t-1}$. By assumption, conditional on $\mathcal{F}_{t-1}$, the random variable ζ is symmetric about zero. Moreover, using the recursion $m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t$, we have

$$g_t - m_t = \beta_1(g_t - m_{t-1}) = \beta_1\zeta.$$

Since $s_t = \beta_2 s_{t-1} + (1-\beta_2)(g_t - m_t)^2$, it follows that

$$s_t = \beta_2 s_{t-1} + (1-\beta_2)\beta_1^2\zeta^2, \qquad \text{and hence} \qquad Y_t = \frac{1}{\sqrt{\beta_2 s_{t-1} + (1-\beta_2)\beta_1^2\zeta^2} + \varepsilon}.$$

Therefore Y_t is an *even* function of ζ, while ζY_t and ζY_t^2 are *odd* functions of ζ. Also,

$$X_t = g_t Y_t = (\zeta + m_{t-1})Y_t.$$

Thus,

$$\begin{aligned}\mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}] &= \mathrm{Cov}[Y_t, (\zeta + m_{t-1})Y_t \mid \mathcal{F}_{t-1}] \\ &= \mathrm{Cov}[Y_t, \zeta Y_t \mid \mathcal{F}_{t-1}] + m_{t-1}\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}]\,.\end{aligned}$$

To handle the first term, note that

$$\mathbb{E}[\zeta Y_t \mid \mathcal{F}_{t-1}] = 0 \quad \text{and} \quad \mathbb{E}[\zeta Y_t^2 \mid \mathcal{F}_{t-1}] = 0$$

because ζY_t and ζY_t^2 are odd in the (conditionally) symmetric variable ζ. Hence

$$\mathrm{Cov}[Y_t, \zeta Y_t \mid \mathcal{F}_{t-1}] = \mathbb{E}[\zeta Y_t^2 \mid \mathcal{F}_{t-1}] - \mathbb{E}[Y_t \mid \mathcal{F}_{t-1}]\,\mathbb{E}[\zeta Y_t \mid \mathcal{F}_{t-1}] = 0,$$

and therefore

$$\mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}] = m_{t-1}\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}]\,.$$

Conclude. Plugging this into equation 4 gives

$$\Delta\mathrm{Var}_t = \beta_1^2 m_{t-1}^2 \mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] + 2\beta_1(1-\beta_1)m_{t-1}^2\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] = (2\beta_1 - \beta_1^2)m_{t-1}^2\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}]\,.$$

Finally, since $\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] \geq 0$ and $2\beta_1 - \beta_1^2 = \beta_1(2-\beta_1) \geq 0$ for $\beta_1 \in [0,1]$, we obtain $\Delta\mathrm{Var}_t \geq 0$, as claimed.

□

B.2 Proof of Proposition 3.2

Proof. We follow the proof of Theorem 3.1 but account for the tracking error. Write $\zeta_t := g_t - m_{t-1} = \xi_t + e$ where $e := \mu_t - m_{t-1}$ with $|e| \leq \delta$, and ξ_t is symmetric about zero. The variance gap identity equation 4 still gives

$$\Delta\mathrm{Var}_t = \beta_1^2 m_{t-1}^2 \mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}] + 2\beta_1(1-\beta_1)m_{t-1}\mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}],$$

where $Y_t = (\sqrt{s_t} + \varepsilon)^{-1}$ and $X_t = g_t Y_t = (\zeta + m_{t-1})Y_t$.

Since $\zeta = \xi_t + e$, the substitution $X_t = (\xi_t + e + m_{t-1})Y_t$ gives

$$\mathrm{Cov}[Y_t, X_t \mid \mathcal{F}_{t-1}] = \mathrm{Cov}[Y_t, \xi_t Y_t \mid \mathcal{F}_{t-1}] + (e + m_{t-1})\mathrm{Var}[Y_t \mid \mathcal{F}_{t-1}].$$

We now bound $m_{t-1}\mathrm{Cov}[Y_t, X_t]$ directly, handling both signs of m_{t-1}. From the decomposition above:

$$m_{t-1}\mathrm{Cov}[Y_t, X_t] = m_{t-1}(e + m_{t-1})\mathrm{Var}[Y_t] + m_{t-1}\mathrm{Cov}[Y_t, \xi_t Y_t].$$

For the first term, $m_{t-1}(e + m_{t-1}) = m_{t-1}^2 + m_{t-1}e$, and by triangle inequality:

$$m_{t-1}(e + m_{t-1}) \geq m_{t-1}^2 - |m_{t-1}||e| \geq m_{t-1}^2 - |m_{t-1}|\delta.$$

This holds regardless of the sign of m_{t-1}, since $m_{t-1}e \geq -|m_{t-1}||e|$ always. Multiplying by $\mathrm{Var}[Y_t] \geq 0$:

$$m_{t-1}(e + m_{t-1})\mathrm{Var}[Y_t] \geq (m_{t-1}^2 - |m_{t-1}|\delta)\mathrm{Var}[Y_t].$$

For the second term, we bound $|m_{t-1}\mathrm{Cov}[Y_t, \xi_t Y_t]| \le |m_{t-1}| \cdot |\mathrm{Cov}[Y_t, \xi_t Y_t]|$. By Cauchy–Schwarz:

$$|\mathrm{Cov}[Y_t, \xi_t Y_t]| \le \sqrt{\mathrm{Var}[Y_t] \cdot \mathrm{Var}[\xi_t Y_t]} \le \sqrt{\mathbb{E}[Y_t^2] \cdot \mathbb{E}[\xi_t^2 Y_t^2]}.$$

Since $Y_t \le 1/\varepsilon$ and $\mathbb{E}[\xi_t^2] \le \sigma_\xi^2$ (bounded by the gradient bound), we obtain:

$$|\mathrm{Cov}[Y_t, \xi_t Y_t]| \le \frac{\sigma_\xi}{\varepsilon^2}.$$

Combining:

$$m_{t-1}\mathrm{Cov}[Y_t, X_t] \ge (m_{t-1}^2 - |m_{t-1}|\delta)\mathrm{Var}[Y_t] - |m_{t-1}| \cdot \frac{\sigma_\xi}{\varepsilon^2}.$$

Plugging into the variance gap identity:

$$\begin{aligned}\Delta\mathrm{Var}_t &\ge \beta_1^2 m_{t-1}^2 \mathrm{Var}[Y_t] + 2\beta_1(1-\beta_1)\big[(m_{t-1}^2 - |m_{t-1}|\delta)\mathrm{Var}[Y_t] - |m_{t-1}|\sigma_\xi/\varepsilon^2\big] \\ &= (2\beta_1 - \beta_1^2)m_{t-1}^2 \mathrm{Var}[Y_t] - 2\beta_1(1-\beta_1)|m_{t-1}|\big[\delta\,\mathrm{Var}[Y_t] + \sigma_\xi/\varepsilon^2\big].\end{aligned}$$

The non-negativity condition $\Delta\mathrm{Var}_t \ge 0$ follows whenever the first term dominates, i.e., when

$$(2\beta_1 - \beta_1^2)|m_{t-1}|\mathrm{Var}[Y_t] \ge 2\beta_1(1-\beta_1)\big[\delta\,\mathrm{Var}[Y_t] + \sigma_\xi/\varepsilon^2\big],$$

which holds provided δ is sufficiently small relative to $|m_{t-1}|$, as stated. □

B.3 Proof of Theorem 3.3

Proof. We work coordinate-wise (all operations are element-wise), so it suffices to consider the scalar case. Let $u \ne 0$ be fixed, $g_0 = Mu$ with $M \gg 1$, and $g_t = u$ for all $t \ge 1$. Assume $\Delta_{-1} = 0$ and $m_{-1} = 0$, and $v_{-1} = s_{-1} = \bar{d} > 0$.

Part I: LaProp and MVN-Grad are uniformly bounded in M. *LaProp (momentum after second-moment normalization).* Define

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2, \qquad z_t^{\mathrm{LP}} := \frac{g_t}{\sqrt{v_t} + \varepsilon}, \qquad \Delta_t^{\mathrm{LP}} := \beta_1 \Delta_{t-1}^{\mathrm{LP}} + (1-\beta_1) z_t^{\mathrm{LP}}.$$

At $t = 0$,

$$v_0 = \beta_2 \bar{d} + (1-\beta_2)M^2u^2 \;\ge\; (1-\beta_2)M^2u^2,$$

hence

$$|z_0^{\mathrm{LP}}| = \frac{|Mu|}{\sqrt{v_0} + \varepsilon} \le \frac{|Mu|}{\sqrt{(1-\beta_2)M^2u^2}} = \frac{1}{\sqrt{1-\beta_2}}.$$

For $t \ge 1$, since $\sqrt{v_t} + \varepsilon \ge \varepsilon$ and $g_t = u$, we have

$$|z_t^{\mathrm{LP}}| \le \frac{|u|}{\varepsilon}.$$

Define

$$Z_{\mathrm{LP}} := \max\Big\{\frac{1}{\sqrt{1-\beta_2}}, \frac{|u|}{\varepsilon}\Big\}.$$

Then $|z_t^{\mathrm{LP}}| \le Z_{\mathrm{LP}}$ for all $t \ge 0$, and the recursion gives

$$|\Delta_t^{\mathrm{LP}}| \le \beta_1 |\Delta_{t-1}^{\mathrm{LP}}| + (1-\beta_1)Z_{\mathrm{LP}} \le \max\{|\Delta_{t-1}^{\mathrm{LP}}|, Z_{\mathrm{LP}}\}.$$

By induction (using $\Delta_{-1}^{\mathrm{LP}} = 0$), $|\Delta_t^{\mathrm{LP}}| \le Z_{\mathrm{LP}}$ for all $t \ge 0$, which is independent of M.

MVN-Grad (momentum after variance-style normalization). Let

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t, \qquad s_t = \beta_2 s_{t-1} + (1-\beta_2)(g_t - m_t)^2,$$

and define

$$z_t^{\mathrm{MV}} := \frac{g_t}{\sqrt{s_t} + \varepsilon}, \qquad \Delta_t^{\mathrm{MV}} := \beta_1 \Delta_{t-1}^{\mathrm{MV}} + (1-\beta_1) z_t^{\mathrm{MV}}.$$

At $t = 0$, $m_0 = (1 - \beta_1)Mu$ so $g_0 - m_0 = \beta_1 Mu$, hence

$$s_0 = \beta_2 \bar{d} + (1 - \beta_2)\beta_1^2 M^2 u^2 \;\geq\; (1 - \beta_2)\beta_1^2 M^2 u^2,$$

and therefore

$$|z_0^{\mathrm{MV}}| = \frac{|Mu|}{\sqrt{s_0} + \varepsilon} \leq \frac{|Mu|}{\sqrt{(1 - \beta_2)\beta_1^2 M^2 u^2}} = \frac{1}{\beta_1 \sqrt{1 - \beta_2}}.$$

For $t \geq 1$, again $\sqrt{s_t} + \varepsilon \geq \varepsilon$ and $g_t = u$, giving

$$|z_t^{\mathrm{MV}}| \leq \frac{|u|}{\varepsilon}.$$

Define

$$Z_{\mathrm{MV}} := \max\Big\{\frac{1}{\beta_1\sqrt{1 - \beta_2}}, \frac{|u|}{\varepsilon}\Big\}.$$

The same induction argument yields $|\Delta_t^{\mathrm{MV}}| \leq Z_{\mathrm{MV}}$ for all $t \geq 0$. Thus we may take $C := \max\{Z_{\mathrm{LP}}, Z_{\mathrm{MV}}\}$, proving the uniform boundedness for LaProp and MVN-Grad.

Part II: Adam. For Adam, with

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t, \qquad v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2, \qquad \Delta_t^{\mathrm{A}} := \frac{m_t}{\sqrt{v_t} + \varepsilon},$$

the closed forms (using $m_{-1} = 0$, $v_{-1} = \bar{d}$) are

$$m_t = (1 - \beta_1)\beta_1^t Mu + (1 - \beta_1^t)u,$$

$$v_t = \beta_2^{t+1}\bar{d} + (1 - \beta_2)\beta_2^t M^2 u^2 + (1 - \beta_2^t)u^2.$$

Define the index

$$t^\star := \min\Big\{t \geq 1 :\; (1 - \beta_2)\beta_2^t M^2 u^2 \leq u^2\Big\}.$$

Then $(1 - \beta_2)\beta_2^{t^\star} M^2 u^2 \leq u^2$ and hence

$$v_{t^\star} \leq \beta_2^{t^\star+1}\bar{d} + u^2 + (1 - \beta_2^{t^\star})u^2 \leq \bar{d} + 2u^2.$$

Consequently,

$$\sqrt{v_{t^\star}} + \varepsilon \leq \sqrt{\bar{d} + 2u^2} + \varepsilon =: D,$$

where D depends only on $(\beta_2, \bar{d}, u, \varepsilon)$. On the other hand, since $M > 0$ and $u \neq 0$, both terms in $m_{t^\star} = (1 - \beta_1)\beta_1^{t^\star} Mu + (1 - \beta_1^{t^\star})u$ share the same sign, so

$$|m_{t^\star}| = (1 - \beta_1)\beta_1^{t^\star} M|u| + (1 - \beta_1^{t^\star})|u| \geq (1 - \beta_1)\beta_1^{t^\star} M|u|.$$

Therefore,

$$|\Delta_{t^\star}^{\mathrm{A}}| = \frac{|m_{t^\star}|}{\sqrt{v_{t^\star}} + \varepsilon} \geq \frac{(1 - \beta_1)\beta_1^{t^\star} M|u|}{D}.$$

This shows that Adam retains a spike-dependent carry-over term of order $(1 - \beta_1)\beta_1^{t^\star} M$. □

B.4 Proof of Corollary 3.3.1

Proof. For LaProp, since $v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2 \geq (1 - \beta_2)g_t^2$, any spike $g_t = M$ yields a normalized input $|z_t| \leq M/(\sqrt{(1 - \beta_2)M^2} + \varepsilon) \leq 1/\sqrt{1 - \beta_2}$, uniformly in M.

For MVN-Grad, the variance proxy uses $s_t = \beta_2 s_{t-1} + (1 - \beta_2)\beta_1^2(g_t - m_{t-1})^2$, so $|z_t| = |g_t|/(\sqrt{s_t} + \varepsilon) \leq |g_t|/(\beta_1\sqrt{1 - \beta_2}\,|g_t - m_{t-1}| + \varepsilon)$. For any spike satisfying $|g_t| \geq 2|m_{t-1}|$, we have $|g_t - m_{t-1}| \geq |g_t|/2$, giving $|z_t| \leq 2/(\beta_1\sqrt{1 - \beta_2})$. Since the momentum buffer u_t is a convex combination of the normalized inputs, any such spike contributes a bounded amount to the update. Thus, for MVN-Grad, spike responses remain uniformly bounded for any sequence of spikes satisfying $|g_t| \geq 2|m_{t-1}|$ at the spike times. □

B.5 Proof of Theorem 3.4

In this subsection we prove Theorem 3.4. Recall that Assumptions 3.1–3.2 motivate the limiting recursions analyzed below. We work under the following oracle model.

Assumption B.1. *Let F be L-smooth and bounded below by $F_\star$ on $\mathbb{R}^d$, and suppose Assumptions 3.1–3.2 hold. In addition, assume a high-SNR stochastic oracle: we observe $g_t = \nabla F(x_{t-1}) + \xi_t$ such that $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0$ and, for some $\delta \in (0,1)$,*

$$|\xi_{t,i}| \le \delta\,|\nabla_i F(x_{t-1})| \quad \textit{a.s.}$$

for all $i \in [d]$ and $t \ge 1$.

The proof proceeds in two steps: we first establish the convergence rate for MVN-Grad under the above assumptions, and then prove the corresponding lower bound for LaProp.

MVN-Grad: To prove the MVN-Grad part of Theorem 3.4, we first establish two auxiliary results. The first shows that, in the low-variance regime where the variance normalizer has stabilized, MVN-Grad reduces to a Heavy-Ball-type recursion driven by the stochastic oracle.

Lemma B.1. *Assume that in a low-variance phase the variance normalizer is locally stabilized with $s_t \approx \sigma^2 \mathbf{1}$ for some $\sigma > 0$. Consider the corresponding stabilized limiting MVN-Grad recursion*

$$z_t = \frac{g_t}{\sigma + \varepsilon_{\text{den}}}, \qquad u_t = \beta_1 u_{t-1} + (1-\beta_1) z_t, \qquad x_t = x_{t-1} - \eta u_t.$$

Eliminating u_{t-1} yields, for all $t \ge 1$,

$$x_t = x_{t-1} + \beta_1(x_{t-1} - x_{t-2}) - \alpha\, g_t, \qquad \alpha := \frac{\eta(1-\beta_1)}{\sigma + \varepsilon_{\text{den}}}, \tag{5}$$

where we take $x_{-1} := x_0$ (equivalently $u_0 := 0$).

Proof. From $x_{t-1} = x_{t-2} - \eta u_{t-1}$ we have $u_{t-1} = -(x_{t-1} - x_{t-2})/\eta$. Using $u_t = \beta_1 u_{t-1} + (1-\beta_1)\frac{g_t}{\sigma+\varepsilon_{\text{den}}}$ gives

$$x_t = x_{t-1} - \eta u_t = x_{t-1} + \beta_1(x_{t-1} - x_{t-2}) - \eta(1-\beta_1)\frac{g_t}{\sigma + \varepsilon_{\text{den}}},$$

which is equation 5. □

Lemma B.1 reduces the analysis of MVN-Grad in the stabilized low-variance regime to a Heavy-Ball recursion with an additive stochastic driving term. The next lemma introduces a suitable potential function and establishes a one-step descent inequality, which we will later telescope to obtain the claimed rate.

Lemma B.2. *Define the increments $y_t := x_t - x_{t-1}$ for $t \ge 0$ with $y_0 := 0$, and set $\alpha := \frac{\eta(1-\beta_1)}{\sigma+\varepsilon_{\text{den}}}$. Assume the stepsize condition*

$$\alpha \le \frac{1-\beta_1^2}{L}. \tag{6}$$

Define the potential

$$\Phi_t \;:=\; F(x_t) + \frac{\beta_1^2}{2\alpha}\,\|y_t\|^2.$$

Then for all $t \ge 1$,

$$\mathbb{E}[\Phi_t \mid \mathcal{F}_{t-1}] \;\le\; \Phi_{t-1} - \frac{\alpha}{2}\,\|\nabla F(x_{t-1})\|^2 + \frac{\alpha}{2}\,\mathbb{E}[\|\xi_t\|^2 \mid \mathcal{F}_{t-1}]. \tag{7}$$

Proof. By L-smoothness,

$$F(x_t) \le F(x_{t-1}) + \langle \nabla F(x_{t-1}), y_t\rangle + \frac{L}{2}\|y_t\|^2. \tag{8}$$

From the definition of y_t and equation 5,

$$y_t + \alpha \nabla F(x_{t-1}) = \beta_1 y_{t-1} - \alpha \xi_t.$$

Taking squared norms and expanding both sides gives

$$\|y_t\|^2 + 2\alpha\langle \nabla F(x_{t-1}), y_t\rangle + \alpha^2\|\nabla F(x_{t-1})\|^2 = \beta_1^2\|y_{t-1}\|^2 - 2\alpha\beta_1\langle y_{t-1}, \xi_t\rangle + \alpha^2\|\xi_t\|^2.$$

Rearranging,

$$2\alpha\langle \nabla F(x_{t-1}), y_t\rangle = \beta_1^2\|y_{t-1}\|^2 - \|y_t\|^2 - \alpha^2\|\nabla F(x_{t-1})\|^2 + \alpha^2\|\xi_t\|^2 - 2\alpha\beta_1\langle y_{t-1}, \xi_t\rangle.$$

Taking conditional expectation given $\mathcal{F}_{t-1}$ and using $\mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0$ (hence $\mathbb{E}[\langle y_{t-1}, \xi_t\rangle \mid \mathcal{F}_{t-1}] = 0$) yields

$$2\alpha\,\mathbb{E}[\langle \nabla F(x_{t-1}), y_t\rangle \mid \mathcal{F}_{t-1}] = \beta_1^2\|y_{t-1}\|^2 - \mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}] - \alpha^2\|\nabla F(x_{t-1})\|^2 + \alpha^2\mathbb{E}[\|\xi_t\|^2 \mid \mathcal{F}_{t-1}]. \tag{9}$$

Taking conditional expectation in equation 8 and substituting equation 9 gives

$$\begin{aligned}
\mathbb{E}[F(x_t) \mid \mathcal{F}_{t-1}] &\le F(x_{t-1}) + \frac{\beta_1^2}{2\alpha}\|y_{t-1}\|^2 - \frac{1}{2\alpha}\mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}] - \frac{\alpha}{2}\|\nabla F(x_{t-1})\|^2 + \frac{\alpha}{2}\mathbb{E}[\|\xi_t\|^2 \mid \mathcal{F}_{t-1}] \\
&\quad + \frac{L}{2}\mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}] \\
&= F(x_{t-1}) + \frac{\beta_1^2}{2\alpha}\|y_{t-1}\|^2 - \frac{\alpha}{2}\|\nabla F(x_{t-1})\|^2 + \frac{\alpha}{2}\mathbb{E}[\|\xi_t\|^2 \mid \mathcal{F}_{t-1}] \\
&\quad - \left(\frac{1}{2\alpha} - \frac{L}{2}\right)\mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}].
\end{aligned}$$

Adding $\frac{\beta_1^2}{2\alpha}\mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}]$ to both sides yields

$$\begin{aligned}
\mathbb{E}\left[F(x_t) + \frac{\beta_1^2}{2\alpha}\|y_t\|^2 \,\middle|\, \mathcal{F}_{t-1}\right] &\le F(x_{t-1}) + \frac{\beta_1^2}{2\alpha}\|y_{t-1}\|^2 - \frac{\alpha}{2}\|\nabla F(x_{t-1})\|^2 + \frac{\alpha}{2}\mathbb{E}[\|\xi_t\|^2 \mid \mathcal{F}_{t-1}] \\
&\quad - \left(\frac{1-\beta_1^2}{2\alpha} - \frac{L}{2}\right)\mathbb{E}[\|y_t\|^2 \mid \mathcal{F}_{t-1}].
\end{aligned}$$

Under equation 6, the last term is nonpositive and can be dropped. Recognizing the definition of Φ_t completes the proof. □

Lemma B.2 introduced one-step descent inequality. Note that by Assumption B.1, we have a.s. $\|\xi_t\|^2 = \sum_{i=1}^d \xi_{t,i}^2 \le \delta^2 \sum_{i=1}^d (\nabla_i F(x_{t-1}))^2 = \delta^2\|\nabla F(x_{t-1})\|^2$. Hence equation 7 reduces to:

$$\mathbb{E}[\Phi_t \mid \mathcal{F}_{t-1}] \le \Phi_{t-1} - \frac{\alpha}{2}(1-\delta^2)\|\nabla F(x_{t-1})\|^2.$$

Taking total expectation gives

$$\mathbb{E}[\Phi_t] \le \mathbb{E}[\Phi_{t-1}] - \frac{\alpha}{2}(1-\delta^2)\mathbb{E}\|\nabla F(x_{t-1})\|^2.$$

Summing from $t = 1$ to T yields

$$\frac{\alpha}{2}(1-\delta^2)\sum_{t=1}^{T}\mathbb{E}\|\nabla F(x_{t-1})\|^2 \le \mathbb{E}[\Phi_0] - \mathbb{E}[\Phi_T] \le \mathbb{E}[\Phi_0] - F_\star.$$

Since $y_0 = 0$, $\Phi_0 = F(x_0)$. Therefore,

$$\frac{\alpha}{2}(1-\delta^2)\sum_{t=1}^{T}\mathbb{E}\|\nabla F(x_{t-1})\|^2 \le F(x_0) - F_\star.$$

Dividing by T gives

$$\min_{0\le t\le T-1}\mathbb{E}\|\nabla F(x_t)\|^2 \le \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla F(x_t)\|^2 \le \frac{2(F(x_0)-F_\star)}{\alpha(1-\delta^2)\,T},$$

Finally, Jensen implies $\min_{0\le t\le T-1}\mathbb{E}\|\nabla F(x_t)\| \le \sqrt{\min_{0\le t\le T-1}\mathbb{E}\|\nabla F(x_t)\|^2}$, so choosing T as stated ensures the ε-stationarity bound.

LaProp. We now prove the LaProp part of Theorem 3.4. In the small-variance high-signal regime, LaProp's second-moment normalization becomes approximately sign-like (see Section 3.3). Accordingly, we analyze the limiting sign dynamics on a smooth quadratic and show that, under the same high-SNR oracle as Assumption B.1, LaProp requires $\Omega(d/\varepsilon^2)$ iterations to reach an ε-stationary point under the standard nonconvex stepsize scale.

Objective and oracle. Throughout this proof we work with the L-smooth function

$$F(x) = \frac{L}{2}\|x\|_2^2, \qquad x \in \mathbb{R}^d,$$

so that $\nabla F(x) = Lx$ and the unique stationary point is $x^\star = 0$. We use the same stochastic oracle model as in Assumption B.1; in particular,

$$g_t = \nabla F(x_{t-1}) + \xi_t = Lx_{t-1} + \xi_t, \qquad \mathbb{E}[\xi_t \mid \mathcal{F}_{t-1}] = 0, \qquad |\xi_{t,i}| \le \delta\,|\nabla_i F(x_{t-1})| \text{ a.s.}$$

Limiting sign-collapse dynamics. As discussed in Section 3.3, in the high-SNR small-variance regime LaProp reduces to the (stochastic) momentum-smoothed sign recursion

$$x_t \;=\; x_{t-1} - \eta_{t-1}\,u_{t-1}, \qquad u_{t-1} \;:=\; (1-\beta_1)\sum_{k=0}^{t-1}\beta_1^{t-1-k}\,\operatorname{sign}(g_{k+1}) \text{ for } t \ge 1 \text{ and } u_0 = 0, \tag{10}$$

where $\beta_1 \in [0,1)$ and $\{\eta_t\}_{t\ge 0}$ is any positive stepsize sequence.

Lemma B.3. *Assume the oracle of Assumption B.1. If $x_{t-1} \succ 0$, then $\operatorname{sign}(g_t) = \mathbf{1}$ a.s. Consequently, along any horizon for which $x_s \succ 0$ holds for all $s \le T$, the recursion equation 10 coincides (a.s.) with the deterministic sign-collapse recursion*

$$x_t \;=\; x_{t-1} - \eta_{t-1}\,u_t, \qquad u_t \;=\; (1-\beta_1)\sum_{k=0}^{t-1}\beta_1^{t-1-k}\,\mathbf{1} \;=\; (1-\beta_1^t)\,\mathbf{1}. \tag{11}$$

Proof. Fix $t \ge 1$ and coordinate j. If $x_{t-1} \succ 0$, then $x_{t-1,j} > 0$ and $\nabla_j F(x_{t-1}) = Lx_{t-1,j} > 0$. By Assumption B.1,

$$g_{t,j} = Lx_{t-1,j} + \xi_{t,j} \;\ge\; (1-\delta)Lx_{t-1,j} \;>\; 0,$$

hence $\operatorname{sign}(g_{t,j}) = +1$ for all j, i.e., $\operatorname{sign}(g_t) = \mathbf{1}$ a.s. Substituting $\operatorname{sign}(g_{k+1}) = \mathbf{1}$ into equation 10 yields equation 11. □

Lemma B.4. *Fix any horizon $T \ge 1$, any stepsizes $\eta_0, \dots, \eta_{T-1} > 0$, and any $\beta_1 \in [0,1)$. Define*

$$a \;:=\; \sum_{t=0}^{T-1}\eta_t\,(1-\beta_1^{t+1}) \;+\; \frac{\eta_{T-1}}{2}\,(1-\beta_1^T), \qquad x_0 := a\,\mathbf{1} \in \mathbb{R}^d, \tag{12}$$

where $\mathbf{1} = (1,\dots,1)$. Consider the recursion equation 10 on $F(x) = \frac{L}{2}\|x\|^2$ under the oracle of Assumption B.1. Then, for every $t = 0, 1, \dots, T$, the following hold almost surely:

1. ***Positive orthant invariance.*** *$x_t \succ 0$. In particular, $\operatorname{sign}(g_t) = \mathbf{1}$ for all $t \le T$ a.s.*
2. ***Closed form.***

$$u_t = (1-\beta_1^t)\,\mathbf{1}, \qquad x_t = \Big(a - \sum_{k=0}^{t-1}\eta_k(1-\beta_1^{k+1})\Big)\mathbf{1}, \qquad x_T = \frac{\eta_{T-1}}{2}(1-\beta_1^T)\,\mathbf{1}.$$

3. ***Gradient lower bound.***

$$\min_{0\le t\le T}\|\nabla F(x_t)\|_2 \;=\; \|\nabla F(x_T)\|_2 \;=\; \frac{L}{2}(1-\beta_1^T)\,\eta_{T-1}\,\sqrt{d}.$$

Equivalently, to guarantee $\min_{0\le t\le T}\|\nabla F(x_t)\|_2 \le \varepsilon$, it is necessary that

$$\eta_{T-1} \;\le\; \frac{2\varepsilon}{L(1-\beta_1^T)\sqrt{d}}. \tag{13}$$

Proof. **Positive orthant invariance.** By Lemma B.3, as long as $x_k \succ 0$ we have $\operatorname{sign}(g_{k+1}) = \mathbf{1}$ a.s. Assume temporarily that $\operatorname{sign}(g_{k+1}) = \mathbf{1}$ for all $k \le t$. Then

$$u_t = (1-\beta_1)\sum_{k=0}^{t-1}\beta_1^{t-k}\mathbf{1} = (1-\beta_1^t)\,\mathbf{1},$$

and equation 10 becomes

$$x_{t+1} = x_t - \eta_t(1-\beta_1^{t+1})\,\mathbf{1}. \tag{14}$$

Closed form. Unrolling equation 14 gives, for any $t \le T$,

$$x_t = x_0 - \sum_{k=0}^{t-1} \eta_k(1-\beta_1^{k+1})\,\mathbf{1} = \Big(a - \sum_{k=0}^{t-1} \eta_k(1-\beta_1^{k+1})\Big)\mathbf{1}.$$

Using equation 12, for every $t \le T$,

$$a - \sum_{k=0}^{t-1} \eta_k(1-\beta_1^{k+1}) \;\ge\; a - \sum_{k=0}^{T-1} \eta_k(1-\beta_1^{k+1}) = \frac{\eta_{T-1}}{2}(1-\beta_1^T) > 0,$$

so $x_t \succ 0$ for all $t \le T$. Lemma B.3 then validates the constant-sign hypothesis, and the expression for x_T follows by plugging $t = T$.

Gradient lower bound. Since $\nabla F(x) = Lx$,

$$\|\nabla F(x_T)\|_2 = L\|x_T\|_2 = L \cdot \frac{\eta_{T-1}}{2}(1-\beta_1^T)\|\mathbf{1}\|_2 = \frac{L}{2}(1-\beta_1^T)\,\eta_{T-1}\,\sqrt{d}.$$

Moreover, equation 14 decreases each coordinate strictly while staying positive, hence $\|x_t\|_2$ (and thus $\|\nabla F(x_t)\|_2$) is strictly decreasing in t, so the minimum over $t \in \{0,\dots,T\}$ is attained at T. Finally, requiring $\|\nabla F(x_T)\|_2 \le \varepsilon$ yields equation 13. □

Theorem B.5. *Take $\eta_t = \frac{c}{L\sqrt{t+1}}$ for a constant $c > 0$. Then, for every horizon $T \ge 1$, there exists an initialization x_0 (given by equation 12) such that the recursion equation 10 satisfies*

$$\min_{0\le t\le T} \|\nabla F(x_t)\|_2 = \frac{c(1-\beta_1^T)\sqrt{d}}{2\sqrt{T}} \qquad \textit{a.s.}$$

Consequently, whenever $1-\beta_1^T \ge \frac{1}{2}$,

$$\min_{0\le t\le T} \|\nabla F(x_t)\|_2 \ge \frac{1}{4}\frac{c\sqrt{d}}{\sqrt{T}}.$$

Hence, to guarantee $\min_{0\le t\le T}\|\nabla F(x_t)\|_2 \le \varepsilon$, it is necessary that

$$T = \Omega\left(\frac{d}{\varepsilon^2}\right).$$

Proof. Applying Lemma B.4 with $\eta_{T-1} = \frac{c}{L\sqrt{T}}$ gives

$$\min_{0\le t\le T} \|\nabla F(x_t)\|_2 = \frac{c(1-\beta_1^T)\sqrt{d}}{2\sqrt{T}}.$$

If $1-\beta_1^T \ge \frac{1}{2}$, then

$$\min_{0\le t\le T} \|\nabla F(x_t)\|_2 \ge \frac{1}{4}\frac{c\sqrt{d}}{\sqrt{T}}.$$

Requiring the right-hand side to be at most ε yields $T = \Omega(d/\varepsilon^2)$. □

Dependence of the initialization on the horizon. For the stepsize $\eta_t = c/(L\sqrt{t+1})$, the initialization in equation 12 satisfies

$$a = \frac{c}{L}\sum_{j=1}^{T}\frac{1-\beta_1^j}{\sqrt{j}} + \frac{c(1-\beta_1^T)}{2L\sqrt{T}} = \Theta\left(\frac{c\sqrt{T}}{L}\right),$$

where the constants may depend on $\beta_1 < 1$. Therefore,

$$F(x_0) - F_\star = \frac{L}{2}\|x_0\|^2 = \frac{Ld}{2}a^2 = \Theta\left(\frac{c^2 dT}{L}\right).$$

Thus, the construction is a finite-horizon lower bound with a horizon-dependent initialization, rather than a lower bound with a uniformly bounded initial objective gap.

B.6 Proof of Theorem 3.5

Proof overview. The proof relies on three ingredients. First, we use a *virtual-iterate reduction*: defining $\tilde{x}_t := x_t - \frac{\beta_1}{1-\beta_1}\eta\, u_t$ decouples the momentum recursion and yields the simple update $\tilde{x}_t - \tilde{x}_{t-1} = -\eta z_t$.

Thus, MVN-Grad with heavy-ball momentum can be analyzed as a momentum-free adaptive method on the virtual sequence, after which the resulting bound is transferred back to the actual iterates x_t.

Second, we use a *covariance decomposition* to handle the dependence between the stochastic gradient and the variance-based normalizer. Writing

$$\mathbb{E}[z_{t,i} \mid \mathcal{F}_{t-1}] = \nabla_i F(x_{t-1})\, \mathbb{E}[w_{t,i} \mid \mathcal{F}_{t-1}] + \mathrm{Cov}(\xi_{t,i}, w_{t,i} \mid \mathcal{F}_{t-1}),$$

with $w_{t,i} = (\sqrt{s_{t,i}}+\varepsilon)^{-1}$, isolates the numerator–denominator coupling into a covariance term. We bound this term using the concavity of the square root, in particular $|\sqrt{a} - \sqrt{b}| \le \sqrt{|a-b|}$, which avoids singular behavior when s_t is small.

Third, Young's inequality controls both the covariance residual and the mismatch between gradients evaluated at the virtual and actual iterates. This yields an inner-product lower bound of the form

$$\langle \nabla F(\tilde{x}_{t-1}), \mathbb{E}_{t-1}[z_t] \rangle \ge \frac{1}{2} S_t - \frac{3}{2} R_t - \frac{3}{2} Q_t,$$

where S_t is the weighted gradient norm, R_t is the virtual-iterate mismatch error, and Q_t is the covariance residual. Combining this lower bound with smoothness of F and telescoping the resulting descent inequality proves Theorem 3.5.

Proof. **Step 1: Virtual iterate reduction.** Define $\tilde{x}_t := x_t - \frac{\beta_1}{1-\beta_1}\eta\, u_t$. Since $u_0 = 0$, we have $\tilde{x}_0 = x_0$. We claim $\tilde{x}_t - \tilde{x}_{t-1} = -\eta z_t$ for all $t \ge 1$. Indeed, from $x_t = x_{t-1} - \eta u_t$ and $u_t = \beta_1 u_{t-1} + (1-\beta_1) z_t$:

$$\begin{aligned}
\tilde{x}_t - \tilde{x}_{t-1} &= (x_t - x_{t-1}) - \frac{\beta_1 \eta}{1-\beta_1}(u_t - u_{t-1}) \\
&= -\eta u_t - \frac{\beta_1 \eta}{1-\beta_1}\big[(\beta_1 - 1)u_{t-1} + (1-\beta_1) z_t\big] \\
&= -\eta u_t + \beta_1 \eta u_{t-1} - \beta_1 \eta z_t \\
&= -\eta\big[\beta_1 u_{t-1} + (1-\beta_1) z_t\big] + \beta_1 \eta u_{t-1} - \beta_1 \eta z_t \\
&= -\eta z_t.
\end{aligned}$$

Step 2: Bounded normalized gradient. Since $\sqrt{s_{t,i}} + \varepsilon \ge \varepsilon$ and $|g_{t,i}| \le G$:

$$|z_{t,i}| = \frac{|g_{t,i}|}{\sqrt{s_{t,i}} + \varepsilon} \le \frac{G}{\varepsilon}, \qquad \|z_t\|^2 \le \frac{dG^2}{\varepsilon^2}.$$

Step 3: Descent on virtual iterates. By L-smoothness applied to the virtual iterates:

$$\begin{aligned}
F(\tilde{x}_t) &\le F(\tilde{x}_{t-1}) + \langle \nabla F(\tilde{x}_{t-1}), \tilde{x}_t - \tilde{x}_{t-1} \rangle + \frac{L}{2}\|\tilde{x}_t - \tilde{x}_{t-1}\|^2 \\
&= F(\tilde{x}_{t-1}) - \eta \langle \nabla F(\tilde{x}_{t-1}), z_t \rangle + \frac{L\eta^2}{2}\|z_t\|^2.
\end{aligned}$$

Step 4: Inner product in expectation. Write $\mathbb{E}_{t-1}[\cdot] := \mathbb{E}[\cdot \mid \mathcal{F}_{t-1}]$ and $\mathrm{Cov}_{t-1}(\cdot,\cdot) := \mathrm{Cov}(\cdot,\cdot \mid \mathcal{F}_{t-1})$. Since $\tilde{x}_{t-1}$ is $\mathcal{F}_{t-1}$-measurable, taking conditional expectations in Step 3 gives

$$\mathbb{E}_{t-1}[F(\tilde{x}_t)] \le F(\tilde{x}_{t-1}) - \eta\, \langle \nabla F(\tilde{x}_{t-1}),\, \mathbb{E}_{t-1}[z_t] \rangle + \frac{L\eta^2}{2}\mathbb{E}_{t-1}[\|z_t\|^2].$$

Let

$$w_{t,i} := \frac{1}{\sqrt{s_{t,i}} + \varepsilon}, \qquad W_{t,i} := \mathbb{E}_{t-1}[w_{t,i}], \qquad c_{t,i} := \mathrm{Cov}_{t-1}(\xi_{t,i}, w_{t,i}).$$

Then $z_{t,i} = g_{t,i} w_{t,i}$ and, since $g_{t,i} = \nabla_i F(x_{t-1}) + \xi_{t,i}$ with $\mathbb{E}_{t-1}[\xi_{t,i}] = 0$, we have

$$\mathbb{E}_{t-1}[z_{t,i}] = \nabla_i F(x_{t-1}) W_{t,i} + c_{t,i}. \tag{15}$$

We first record simple bounds on $W_{t,i}$. Since $\|g_t\|_\infty \le G$ a.s., we have $\|\nabla F(x_{t-1})\|_\infty \le G$. Moreover, m_t is an EMA of bounded gradients, so $\|m_t\|_\infty \le G$. Hence $|g_{t,i} - m_{t,i}| \le 2G$ and therefore $0 \le s_{t,i} \le 4G^2$ a.s. Thus

$$\frac{1}{2G+\varepsilon} \le w_{t,i} \le \frac{1}{\varepsilon}, \qquad \frac{1}{2G+\varepsilon} \le W_{t,i} \le \frac{1}{\varepsilon}. \tag{16}$$

We next bound the covariance term. Define

$$a_{t,i} := \nabla_i F(x_{t-1}), \qquad \delta_{t,i} := a_{t,i} - m_{t-1,i},$$

and let

$$\bar{s}_{t,i} := \beta_2 s_{t-1,i} + (1-\beta_2)\beta_1^2 \delta_{t,i}^2, \qquad \bar{w}_{t,i} := \frac{1}{\sqrt{\bar{s}_{t,i}} + \varepsilon}.$$

Here $\bar{s}_{t,i}$ is the value of $s_{t,i}$ at $\xi_{t,i} = 0$, and $\bar{w}_{t,i}$ is $\mathcal{F}_{t-1}$-measurable. Since $\mathbb{E}_{t-1}[\xi_{t,i}\bar{w}_{t,i}] = 0$,

$$c_{t,i} = \mathbb{E}_{t-1}[\xi_{t,i}(w_{t,i} - \bar{w}_{t,i})]\,.$$

Also,

$$|w_{t,i} - \bar{w}_{t,i}| \le \frac{|\sqrt{s_{t,i}} - \sqrt{\bar{s}_{t,i}}|}{\varepsilon^2} \le \frac{\sqrt{|s_{t,i} - \bar{s}_{t,i}|}}{\varepsilon^2}.$$

Using

$$s_{t,i} - \bar{s}_{t,i} = (1-\beta_2)\beta_1^2(2\xi_{t,i}\delta_{t,i} + \xi_{t,i}^2),$$

we get

$$|w_{t,i} - \bar{w}_{t,i}| \le \frac{\beta_1\sqrt{1-\beta_2}}{\varepsilon^2}\sqrt{|2\xi_{t,i}\delta_{t,i} + \xi_{t,i}^2|}.$$

Therefore, by Cauchy–Schwarz,

$$|c_{t,i}| \le \frac{\beta_1\sqrt{1-\beta_2}}{\varepsilon^2}\sqrt{\mathbb{E}_{t-1}[\xi_{t,i}^2]\,\mathbb{E}_{t-1}[|2\xi_{t,i}\delta_{t,i} + \xi_{t,i}^2|]}.$$

Since $|a_{t,i}| \le G$, $|m_{t-1,i}| \le G$, and $|g_{t,i}| \le G$, we have $|\delta_{t,i}| \le 2G$ and $|\xi_{t,i}| \le 2G$ almost surely. Thus the variance proxy can be bounded using $\bar{\sigma}_\xi := \min\{\sigma_\xi, 2G\}$. Since the final bound is monotone in σ_ξ, we relabel $\bar{\sigma}_\xi$ as σ_ξ and assume $\sigma_\xi \le 2G$ without loss of generality.

$$\mathbb{E}_{t-1}[|2\xi_{t,i}\delta_{t,i} + \xi_{t,i}^2|] \le 4G\,\mathbb{E}_{t-1}[|\xi_{t,i}|] + \mathbb{E}_{t-1}[\xi_{t,i}^2] \le 4G\sigma_\xi + \sigma_\xi^2 \le 6G\sigma_\xi.$$

Thus

$$|c_{t,i}| \le \frac{\beta_1\sqrt{6G}\,(1-\beta_2)^{1/2}\sigma_\xi^{3/2}}{\varepsilon^2}. \tag{17}$$

Now we lower bound the descent inner product. Let

$$a_t := \nabla F(x_{t-1}), \qquad b_t := \nabla F(\tilde{x}_{t-1}), \qquad e_t := b_t - a_t.$$

Using equation 15,

$$\langle b_t, \mathbb{E}_{t-1}[z_t]\rangle = \sum_i (a_{t,i} + e_{t,i})(a_{t,i}W_{t,i} + c_{t,i}).$$

Define

$$S_t := \sum_i a_{t,i}^2 W_{t,i}, \qquad R_t := \sum_i e_{t,i}^2 W_{t,i}, \qquad Q_t := \sum_i \frac{c_{t,i}^2}{W_{t,i}}.$$

Then, by Young's inequality,

$$\sum_i e_{t,i} a_{t,i} W_{t,i} \ge -\frac{1}{4}S_t - R_t,$$

$$\sum_i a_{t,i} c_{t,i} \ge -\frac{1}{4}S_t - Q_t,$$

and

$$\sum_i e_{t,i} c_{t,i} \geq -\frac{1}{2} R_t - \frac{1}{2} Q_t.$$

Combining these three bounds gives

$$\langle b_t, \mathbb{E}_{t-1}[z_t] \rangle \geq \frac{1}{2} S_t - \frac{3}{2} R_t - \frac{3}{2} Q_t. \tag{18}$$

It remains to bound R_t and Q_t. By L-smoothness,

$$\|e_t\| \leq L \|\tilde{x}_{t-1} - x_{t-1}\| = \frac{L\beta_1 \eta}{1-\beta_1} \|u_{t-1}\|.$$

Since u_{t-1} is an EMA of vectors z_k satisfying $\|z_k\| \leq G\sqrt{d}/\varepsilon$, we have $\|u_{t-1}\| \leq G\sqrt{d}/\varepsilon$. Therefore,

$$\|e_t\| \leq \frac{L\beta_1 \eta G \sqrt{d}}{(1-\beta_1)\varepsilon}.$$

Using $W_{t,i} \leq 1/\varepsilon$,

$$R_t \leq \frac{1}{\varepsilon} \|e_t\|^2 \leq \frac{L^2 \beta_1^2 \eta^2 d G^2}{(1-\beta_1)^2 \varepsilon^3}. \tag{19}$$

Similarly, using equation 16 and equation 17,

$$Q_t \leq \frac{6\beta_1^2 d(1-\beta_2) G(2G+\varepsilon)\sigma_\xi^3}{\varepsilon^4}. \tag{20}$$

In particular, if $\varepsilon \leq G$, then

$$Q_t \leq \frac{18\beta_1^2 d(1-\beta_2) G^2 \sigma_\xi^3}{\varepsilon^4}.$$

Substituting equation 18, equation 19, and equation 20 into the conditional descent inequality, and using $\mathbb{E}_{t-1}[\|z_t\|^2] \leq dG^2/\varepsilon^2$, gives

$$\begin{aligned} \mathbb{E}_{t-1}[F(\tilde{x}_t)] &\leq F(\tilde{x}_{t-1}) - \frac{\eta}{2} S_t + \frac{3\eta}{2} R_t + \frac{3\eta}{2} Q_t + \frac{L\eta^2 d G^2}{2\varepsilon^2} \\ &\leq F(\tilde{x}_{t-1}) - \frac{\eta}{2} S_t + \frac{3L^2\beta_1^2\eta^3 d G^2}{2(1-\beta_1)^2\varepsilon^3} + \frac{9\beta_1^2 \eta d(1-\beta_2) G(2G+\varepsilon)\sigma_\xi^3}{\varepsilon^4} + \frac{L\eta^2 d G^2}{2\varepsilon^2}. \end{aligned} \tag{21}$$

Finally, by equation 16,

$$S_t = \sum_i \nabla_i F(x_{t-1})^2 W_{t,i} \geq \frac{1}{2G+\varepsilon} \|\nabla F(x_{t-1})\|^2.$$

Step 5: Telescope. Taking total expectations in equation 21 and summing from $t = 1$ to T, we obtain

$$\begin{aligned} \frac{\eta}{2} \sum_{t=1}^{T} \mathbb{E}[S_t] &\leq F(\tilde{x}_0) - \mathbb{E}[F(\tilde{x}_T)] + \frac{3L^2\beta_1^2\eta^3 T d G^2}{2(1-\beta_1)^2\varepsilon^3} \\ &\quad + \frac{9\beta_1^2 \eta T d(1-\beta_2) G(2G+\varepsilon)\sigma_\xi^3}{\varepsilon^4} + \frac{L\eta^2 T d G^2}{2\varepsilon^2}. \end{aligned}$$

Since $\tilde{x}_0 = x_0$ and $F(\tilde{x}_T) \geq F_\star$, this gives

$$\begin{aligned} \frac{\eta}{2} \sum_{t=1}^{T} \mathbb{E}[S_t] &\leq \Delta_0 + \frac{3L^2\beta_1^2\eta^3 T d G^2}{2(1-\beta_1)^2\varepsilon^3} \\ &\quad + \frac{9\beta_1^2 \eta T d(1-\beta_2) G(2G+\varepsilon)\sigma_\xi^3}{\varepsilon^4} + \frac{L\eta^2 T d G^2}{2\varepsilon^2}. \end{aligned}$$

Moreover, by equation 16,

$$S_t = \sum_i \nabla_i F(x_{t-1})^2 W_{t,i} \geq \frac{1}{2G+\varepsilon} \|\nabla F(x_{t-1})\|^2.$$

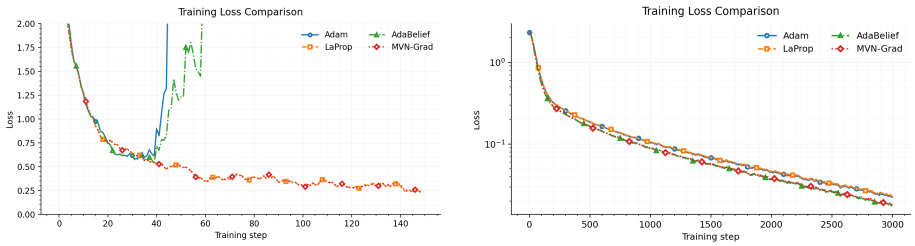
Therefore,

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(x_t)\|^2 \leq \frac{2(2G+\varepsilon)\Delta_0}{\eta T} + (2G+\varepsilon)L\eta\frac{dG^2}{\varepsilon^2} + \frac{3(2G+\varepsilon)L^2\beta_1^2\eta^2 dG^2}{(1-\beta_1)^2\varepsilon^3} + \frac{18\beta_1^2 d(1-\beta_2)G(2G+\varepsilon)^2\sigma_\xi^3}{\varepsilon^4}.$$

Under $\varepsilon \leq G$, we have $2G+\varepsilon \leq 3G$. Also, by the stepsize condition $\eta \leq c\frac{(1-\beta_1)^2\varepsilon}{L\beta_1^2}$, for $c > 0$ sufficiently small, the third term is dominated by the second term. Hence there exist constants $C_1, C_2 > 0$, depending only on β_1, such that

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla F(x_t)\|^2 \leq \frac{C_1 G\Delta_0}{\eta T} + C_1 L\eta\frac{dG^3}{\varepsilon^2} + \frac{C_2 d(1-\beta_2)G^3\sigma_\xi^3}{\varepsilon^4}.$$

This proves equation 3. □



(a) Batch size 128 (b) Batch size 1024

Figure 3: MNIST training loss (2-layer MLP). (a) At small batch size, MVN-Grad produces smoother dynamics with fewer loss spikes. (b) At large batch size (low-variance), variance normalization preserves gradient scale and yields faster convergence. Hyperparameters in Appendix C.2.

C Experimental Details

C.1 Compute Resources

CIFAR-100/ResNet-18 experiments were run on NVIDIA H100 NVL and NVIDIA GH200 GPUs, depending on availability. The batch-size 128 runs used 1 NVIDIA H100 NVL GPU and took approximately 4000 seconds per run. The batch-size 1024 runs took approximately 4000–4300 seconds per completed run, using either 1 NVIDIA H100 NVL GPU or 1 NVIDIA GH200 GPU.

WikiText-103/GPT-2 30M experiments were run on NVIDIA H100 GPUs. Each run used batch size 128, sequence length 512, and 100k training steps, and took approximately 4 wall-clock hours based on the logged throughput.

OpenWebText/GPT-2 124M experiments were run on NVIDIA GH200 GPUs. The selected-configuration runs took approximately 46k–48k seconds per run, corresponding to roughly 13 wall-clock hours. MVN-GradW has one additional optimizer-state tensor relative to AdamW / AdaBeliefW / LaPropW; in our OpenWebText runs, wall-clock time remained comparable across optimizers. The full research process required additional compute beyond the final reported runs due to pilot experiments, failed configurations, and hyperparameter sweeps.

FineWeb/GPT-style 350M experiments were run on NVIDIA GH200 GPUs. Selected-configuration runs took approximately 2.5 wall-clock hours per run.

C.2 MNIST Experiment Hyperparameters

Fig. 3a.

$$\eta = 5 \cdot 10^{-3}, \qquad \beta_1 = 0.999, \qquad \beta_2 = 0.7, \qquad \varepsilon = 10^{-8}, \qquad \text{batch size} = 128.$$

Fig. 3b.

$$\eta = 1 \cdot 10^{-4}, \qquad \beta_1 = 0.95, \qquad \beta_2 = 0.999, \qquad \varepsilon = 10^{-8}, \qquad \text{batch size} = 1024.$$

No optimizer-specific tuning was performed in either panel.

C.3 MNIST Conditional-Variance Diagnostic

For Figure 4, we estimate the one-step conditional update variance along a real training trajectory. We train a two-hidden-layer MLP on MNIST using AdaBelief and MVN-Grad with batch size 16, learning rate 10^{-3}, $\beta_1 = 0.9$, and $\beta_2 = 0.95$. At each selected checkpoint, we freeze the optimizer state $(x_{t-1}, m_{t-1}, s_{t-1})$ and evaluate both methods from the same state by drawing $K = 128$ independent mini-batches. Using the resulting gradients, we form one-step updates and estimate

$$\Delta\mathrm{Var}(t) = \mathrm{Var}\big(\Delta_t^{\mathrm{AB}} \mid \mathcal{F}_{t-1}\big) - \mathrm{Var}\big(\Delta_t^{\mathrm{MV}} \mid \mathcal{F}_{t-1}\big)$$

via Monte Carlo. Variances are computed coordinate-wise, averaged over parameters, and then averaged over three seeds.

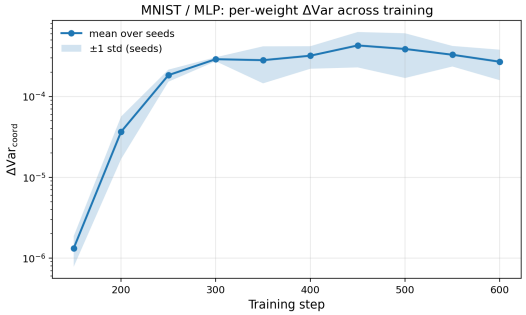
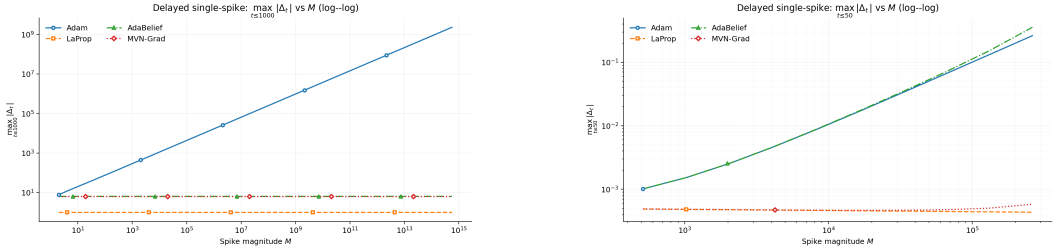


Figure 4: Conditional update-variance gap via Monte Carlo at frozen checkpoints.



(a) $T = 1000;\ \beta_1 = 0.9,\ \beta_2 = 0.6.$ (b) $T = 50;\ \beta_1 = 0.99999,\ \beta_2 = 0.1.$

Figure 5: Delayed single-spike robustness: peak update magnitude $\max_{0\le\tau\le T}|\Delta_\tau|$ versus spike size M. Within each panel, all optimizers use the same hyperparameters. Details in Appendix C.4.

C.4 Single-spike robustness experiment details

Figures 5a–5b simulate the single-spike model from Theorem 3.3. We set $g_0 = Mu$ for fixed $u \neq 0$, followed by $g_t = u$ for all $t \geq 1$, and initialize the adaptive statistics with $m_{-1} = 0$ and $v_{-1} = s_{-1} = \bar{d} > 0$. For each configuration, we simulate T iterations and record the peak update magnitude

$$\max_{0\le\tau\le T}|\Delta_\tau|$$

as a function of the spike size M. Within each panel, all optimizers use identical hyperparameters; only M varies across runs. A formal spike analysis for AdaBelief is more involved because its variance estimator is coupled to the momentum state, so we report its behavior empirically.

Figure 5a (long horizon).

$$\beta_1 = 0.9, \qquad \beta_2 = 0.6, \qquad \varepsilon = 10^{-8}, \qquad \bar{d} = 1, \qquad u = 10^{-3}, \qquad T = 1000.$$

Figure 5b (early carry-over regime).

$$\beta_1 = 0.99999, \qquad \beta_2 = 0.1, \qquad \varepsilon = 10^{-8}, \qquad \bar{d} = 10, \qquad u = 10, \qquad T = 50.$$

No optimizer-specific tuning was performed in either panel.

C.5 CIFAR-100 with ResNet-18

Hyperparameter search. We apply the same shared search grids to all methods at each batch size; variance-normalized methods additionally sweep ε_s. For batch size 1024, we first swept $\eta \in 2\times10^{-3}, 3\times10^{-3}, 4\times10^{-3}$ with $\beta_1 = 0.9$ and $\beta_2 \in 0.99, 0.999$. We then swept $\beta_1 \in 0.7, 0.8, 0.9, 0.95$, and $\beta_2 \in 0.99, 0.999$ at $\eta \in 10^{-4}, 10^{-3}, 4\times10^{-3}, 10^{-2}$. For batch size 128, we swept $\eta \in \{10^{-4}, 5\times10^{-4}, 10^{-3}\}$ with $\beta_1 = 0.9$ and $\beta_2 \in \{0.99, 0.999\}$; for AdaBelief and MVN-Grad we additionally evaluate $\varepsilon_s \in \{10^{-6}, 10^{-8}\}$, while using $\varepsilon = 10^{-8}$ for all methods. We then swept $\beta_1 \in \{0.7, 0.8, 0.9, 0.95\}$ over $\eta \in \{5\times10^{-4}, 10^{-3}, 5\times10^{-3}, 10^{-2}\}$. In both cases, we perform single-seed selection followed by 3-seed evaluation.

Table 4: Selected optimizer hyperparameters for CIFAR-100 / ResNet-18 (batch size 128).

Optimizer	η	β_1	β_2	ε_s
LaProp	5×10^{-4}	0.7	0.999	-
AdaBelief	1×10^{-3}	0.9	0.99	10^{-8}
Adam	5×10^{-4}	0.8	0.999	-
MVN-Grad	1×10^{-3}	0.95	0.99	10^{-8}

Table 5: Selected optimizer hyperparameters for CIFAR-100 / ResNet-18 (batch size 1024).

Optimizer	η	β_1	β_2	ε_s
AdaBelief	4×10^{-3}	0.7	0.99	10^{-8}
Adam	1×10^{-3}	0.8	0.999	-
LaProp	1×10^{-3}	0.95	0.999	-
MVN-Grad	4×10^{-3}	0.7	0.99	10^{-8}

Selected hyperparameters for CIFAR-100 / ResNet-18 For reproducibility, Tables 4–5 report the configuration selected for each optimizer using the single selection seed described above.

C.6 Language-modeling hyperparameter grids and selected configurations

This appendix lists the tuning grids and selected optimizer hyperparameters corresponding to the language-modeling results in Table 2 and Table 10. All other settings follow the training setup described in the main text.

WikiText-103 sweep. We use a two-stage tuning protocol. First, we perform a coarse sweep over learning rates $\{10^{-4}, 10^{-3}, 10^{-2}\}$ with $\beta_1 \in \{0.9, 0.95\}$ and $\beta_2 \in \{0.99, 0.999\}$ to identify stable learning-rate regimes. Final comparisons use a focused sweep in which all optimizers are evaluated on identical grids with learning rates $\{5 \times 10^{-4}, 10^{-3}\}$, $\beta_1 \in \{0.6, 0.7, 0.9, 0.95\}$, and $\beta_2 \in \{0.95, 0.99\}$. Variance-normalized methods additionally evaluate $\varepsilon_s \in \{0, 10^{-8}\}$.

OpenWebText sweep. Based on preliminary runs at alternative learning rates, we identify $\eta = 10^{-4}$ as a stable common learning-rate regime and then evaluate the same (β_1, β_2) grid for each optimizer. For variance-normalized methods we set $\varepsilon_s = 0$. For each optimizer, we evaluate the same nine representative (β_1, β_2) pairs: $(0.6, 0.99)$, $(0.6, 0.999)$, $(0.7, 0.95)$, $(0.7, 0.99)$, $(0.7, 0.999)$, $(0.8, 0.9)$, $(0.8, 0.99)$, $(0.8, 0.999)$, and $(0.9, 0.95)$. We select the configuration minimizing final validation loss and re-run it with three seeds.

FineWeb sweep. The 350M model has 24 layers, 16 attention heads, embedding dimension 1024, and context length 1024. Runs use micro-batches of size 12 with 72 gradient-accumulation steps, bf16 precision, no dropout, weight decay 0.1, gradient clipping at 1, 400 warmup iterations, and 4,000 training iterations. Validation uses 200 batches every 1,000 iterations.

Preliminary experiments indicated that $(\beta_1, \beta_2) = (0.9, 0.95)$ was a stable common momentum setting across all four optimizers. To keep the larger-scale comparison compute-matched, we fix this pair for every method and evaluate the shared learning-rate grid $\{10^{-3}, 2.5 \times 10^{-3}, 5 \times 10^{-3}\}$, with a final-to-peak learning-rate ratio of 0.1. Thus, the 350M experiment is intended as a controlled scaling comparison rather than an exhaustive optimizer-specific momentum search. For variance-normalized methods, we set $\varepsilon_s = 0$. All methods use the same final-update row-clipping rule with clipping parameter 1. We select by final validation loss using seed 1 and report mean $\pm$ standard deviation over seeds 1–4 at the selected configuration. All four optimizers select $\eta = 2.5 \times 10^{-3}$.

Table 6: WikiText-103 selected optimizer hyperparameters used in Table 2. A dash indicates that the method does not use ε_s.

Optimizer	η	β_1	β_2	ε_s
Adam	1×10^{-3}	0.95	0.99	–
LaProp	1×10^{-3}	0.95	0.99	–
AdaBelief	1×10^{-3}	0.95	0.99	0
MVN-Grad	1×10^{-3}	0.95	0.99	0

Table 7: OpenWebText selected optimizer hyperparameters used in Table 2. A dash indicates that the method does not use ε_s.

Optimizer	η	β_1	β_2	ε_s
AdamW	1×10^{-4}	0.6	0.99	–
LaPropW	1×10^{-4}	0.6	0.999	–
AdaBeliefW	1×10^{-4}	0.6	0.999	0
MVN-GradW	1×10^{-4}	0.6	0.999	0

Table 8: FineWeb selected optimizer hyperparameters. A dash indicates that the method does not use ε_s.

Optimizer	η	β_1	β_2	ε_s
AdamW	2.5×10^{-3}	0.9	0.95	–
LaProp	2.5×10^{-3}	0.9	0.95	–
AdaBelief	2.5×10^{-3}	0.9	0.95	0
MVN-Grad	2.5×10^{-3}	0.9	0.95	0

Table 9: FineWeb 350M seed-1 validation loss over the shared learning-rate grid. All methods use $(\beta_1, \beta_2) = (0.9, 0.95)$ and a final-to-peak learning-rate ratio of 0.1. Bold indicates the best learning rate for each optimizer.

Optimizer	10^{-3}	2.5×10^{-3}	5×10^{-3}
AdamW	3.2651	**3.2308**	3.2480
LaProp	3.2696	**3.2290**	3.2372
AdaBelief	3.2607	**3.2325**	3.2576
MVN-Grad	3.2605	**3.2249**	3.2490

Table 10: Language-modeling training cross-entropy loss (mean $\pm$ standard deviation) for the selected configurations. Results use four seeds for WikiText-103 and FineWeb, and three seeds for OpenWebText. Method names denote optimizer families.

Optimizer	WikiText-103 GPT-2 30M	OpenWebText GPT-2 124M	FineWeb GPT-style 350M
Adam / AdamW	4.267 ± 0.036	2.972 ± 0.009	3.2033 ± 0.0041
LaProp	4.297 ± 0.067	2.972 ± 0.009	3.2039 ± 0.0012
AdaBelief	4.262 ± 0.017	2.956 ± 0.008	3.2026 ± 0.0073
MVN-Grad	4.248 ± 0.047	2.955 ± 0.007	3.2026 ± 0.0069