ADAMW SCALING LAWS FROM FIRST PRINCIPLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Scaling language-model pre-training to larger token budgets and model sizes requires retuning optimizer hyperparameters, yet existing transfer rules are predominantly empirical and offer limited theoretical guidance. We study this problem through the convergence behavior of `SGD` with momentum (`SGDM`) and `AdamW` under smoothness, bounded stochastic-gradient variance, and the μ-Kurdyka–Łojasiewicz condition. For both algorithms, we establish convergence guarantees and derive their optimization error under a fixed token budget T. In the long-horizon regime, our analysis yields token-budget-aware BST scaling rules: the batch size should grow as $B \propto T^{2/3}$, while the learning rate and momentum parameters remain approximately constant. We further prove a matching lower bound for zero-respecting stochastic first-order algorithms, showing that the deterministic and stochastic terms in our upper bounds are unimprovable, demonstrating the tightness of our approach. Experiments on FineWeb with Transformer language models ranging up to 1.4B parameters support these predictions. In particular, after scaling the batch size according to the BST rule, the optimal learning rate transfers across token budgets within each model scale. Together, our results provide a theoretically grounded prescription for transferring hyperparameters to longer pre-training runs without requiring expensive empirical fits.

1 INTRODUCTION

We develop a scaling theory for `AdamW` that mathematically characterizes the token budget and hyperparameters needed to reach a target training accuracy. For a network with parameters $x \in \mathbb{R}^d$ and training loss $f(x)$, the optimization problem is $\min_x f(x)$. Given a token budget T and B batches of tokens (B/S sequences of length S), training consists of $K = T/B$ updates. We seek hyperparameters for which the expected loss gap $\mathbb{E}[f(x_K) - f^\star]$ is at most ε, where $f^\star = \inf_x f(x)$.

Existing scaling approaches in this setting fall into two camps. The first camp derives empirical "scaling laws" by fitting loss curves and hyperparameter prescriptions to training runs. For instance, a recent language-model pre-training study trained more than 3,700 models from scratch, consuming nearly one million H800 GPU-hours (Li et al., 2025a). Despite this substantial cost, reported learning-rate and batch-size exponents remain inconsistent across studies (see Table 1). Fitted laws also depend on experimental design and checkpoint selection (McLeish et al., 2025).

The second camp derives parameterizations that preserve stable feature learning as a network grows. The maximal-update parameterization (μP) specifies layerwise initialization and update scalings that enable hyperparameter transfer across widths (Yang & Hu, 2021; Yang et al., 2021). Its foundational analysis takes width to infinity at a fixed training horizon. However, it does not prescribe the training budget required to reach a specified loss gap, and requires the target batch size to be specified separately. These parameterizations depend on the architecture and optimizer. For instance, structured state-space models require a separate construction (Vankadara et al., 2024).

In contrast to the scaling laws camp, our approach is based on convergence analysis rather than curve fitting. Our analysis uses the μ-Kurdyka–Łojasiewicz (μ-KŁ) condition, under which the gradient norm is at least μ times the loss gap (Łojasiewicz, 1963; Kurdyka, 1998). We justify its use in several neural-network settings and analyze optimization trajectories contained in regions where it holds. The resulting guarantees quantify loss reduction over the entire training horizon and complement the fixed-horizon model scaling theory developed in the second camp.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

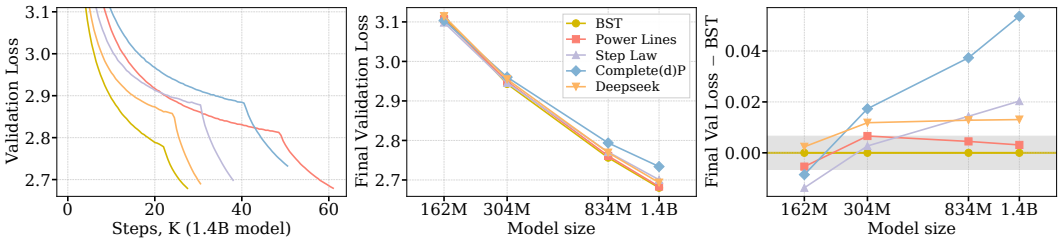


Figure 1: **BST performs competitively with other scaling laws during model size and token budget co-scaling while requiring the fewest steps.** The comparison of scaling laws for AdamW for Peri-LN models across four model sizes at the token budget $T = 20$B on the FineWeb dataset, with width and depth co-scaled using CompleteP at a fixed aspect ratio of 64. **Left:** validation loss dynamics; train loss dynamics are given in Section C.3. **Middle:** final validation loss varying the model size. **Right:** the difference from BST at the same model size. The shaded band is $\pm 2\sigma_\Delta$ with σ_Δ =3.27e-3, the standard deviation of the seed difference of $\text{rule}_i - \text{BST}_i$ over 3 seeds, averaged over the five rules at 162M (see Figure C.4).

Our bounds yield a direct rule for adjusting parameters when increasing the token budget T. In the long-horizon regime, the batch size should be scaled as $B \propto T^{2/3}$, while the learning rate η and momentum parameter α remain constant. For AdamW, the second-moment parameter β also remains constant, while weight-decay should be scaled as $\lambda \propto d^{-1/2}T^{-1/3}$, omitting logarithmic factors. These prescriptions give an expected loss gap ε of order $T^{-1/3}$. Larger batches reduce gradient noise but leave fewer updates under the same token budget. The exponent $2/3$ is precisely the growth rate that keeps these effects balanced.

Islamov et al. (2026) derive the same batch exponent under μ-KŁ for stochastic conditional-gradient methods (SCG), and we retain their batch–sequence–token (BST) name in our work. Their guarantee requires a learning rate that decreases as the token budget grows, while our analysis of SGDM and AdamW allows a constant learning rate. To summarize, our contributions are as follows.

1. **Convergence analysis.** We establish last-iterate convergence guarantees for SGDM and AdamW with stochastic gradients under the μ-KŁ condition, with explicit dependence on batch size and optimizer hyperparameters (Theorems 1 and 2).
2. **Lower bounds.** We construct smooth μ-KŁ instances on which zero-respecting stochastic first-order methods require the same leading dependence on accuracy, smoothness, and noise variance as the corresponding terms in our upper bounds (Theorem 3).
3. **Neural-network geometry.** To support the use of μ-KŁ condition, we derive restricted μ-KŁ guarantees for linear networks, networks with rich last-layer features, and the lazy neural tangent kernel regime (Appendix F).
4. **Scaling rules.** We derive BST scaling rules for batch size, learning rate, momentum, and weight decay hyperparameters of AdamW from the convergence bounds, without fitting the transfer exponents. Our rules prescribe how the hyperparameters should change with token horizon T and model size d.
5. **Empirical validation.** We evaluate batch and learning-rate transfer on FineWeb with Transformer models up to 1.4B parameters, across token budgets and compare BST scaling rules in joint model–budget scaling experiments (Figure 1, Sections 6.2 and 6.4). We show that the derived BST scaling rules for AdamW perform competitively with existing empirically derived scaling laws without fitting training curves and with much fewer steps.

2 Related Works

Empirical Scaling Laws. Empirical scaling laws model performance as a power-law function of dataset size, model size, and training compute (Hestness et al., 2017; Kaplan et al., 2020; Hoffmann et al., 2022b). Their exponents can be sensitive to the training setup and fitting methodology

Table 1: Rules for transferring optimizer hyperparameters to a larger token budget and model size. An entry T^{κ} multiplies the tuned value by $(T_1/T_0)^{\kappa}$ when the budget grows from T_0 to T_1, and likewise for the parameter count d, the non-embedding count d_{ne}, and general compute C. "Const." keeps the tuned value, "Free" leaves it to the user, and "–" marks a hyperparameter the rule does not cover and set to some default value. **Model Scaling** records whether the model is held fixed, varied separately from T, or tied to T; **Type** marks rules fitted to sweeps (Em) or derived from theory (Th).

Rule	Optimizer	Learning Rate	Batch Size	Weight Decay	Mom.	Model Scaling	Type
Bi et al. (2024)[a]	AdamW	$C^{-0.125}$	$C^{0.327}$	–	–	Tied to T	Em
Bjorck et al. (2025)[b]		$d^{-0.23}T^{-0.32}$	Const.	–	–	Separate	Em
Li et al. (2025a)		$d_{\text{ne}}^{-0.713}T^{0.307}$	$T^{0.571}$	–	–	Separate	Em
Bergsma et al. (2025)[c]		Const.	$T^{0.383}$	$d^{-0.527}T^{-0.09}$	–	Separate	Em
Mlodozeniec et al. (2026)[d]		$(B/T)^{1/2}$	Free	$(B/T)^{1/2}$	B/T	Separate	Th
Schaipp (2026) [e]		–	$T^{0.566}$	–	–	Separate	Em
Ajroldi et al. (2026)		$d^{-0.417}T^{0.086}$	$d^{0.145}\,T^{0.460}$	–	–	Separate	Em
BST [Ours]		Const.	$T^{2/3}$	$d^{-1/2}T^{-1/3}$	Const.	Separate	Th
Li et al. (2026) [f]	AdamW/SGDW	$(B/T)^{1/2}$	Free	$(B/T)^{1/2}$	–	Separate	Th
Filatov et al. (2026)	Scion	$T^{-0.28}$	$T^{0.45}$	–	–	Fixed	Em
von Rütte et al. (2026) [g]	LaProp	$T^{0.281}$	$T^{0.823}$	–	–	Separate	Em
Islamov et al. (2026)	SCG	$T^{-1/3}$	$T^{2/3}$	–	Const.	Separate	Th
Shulgin et al. (2026)	uSCG	$T^{-7/12}$	$T^{1/6}$	–	$T^{-1/3}$	Fixed	Th
BST [Ours]	SGDM	Const.	$T^{2/3}$	–	Const.	Separate	Th

[a] One can also fit the compute exponents along the compute-optimal path of the same paper, where $T \propto C^{0.4757}$, obtaining $\eta \propto T^{-0.263}$ and $B \propto T^{0.688}$ at a fixed model size.
[b] In this study, they arrived at the scaling law using a fixed batch size. The fit is restricted to models with at least 760M parameters.
[c] Converted from the fitted timescale $B/(\eta\lambda T) \propto (T/d)^{-0.527}$ at a constant learning rate, which the authors report to be more efficient than adjusting the learning rate.
[d] The scaling law for the batch size is not specified and treated as a user choice. The momentum column refers to $1-\beta_{1,2}$, and width and depth have separate multipliers. Bethune et al. (2026) replace B/T by B/T^{γ} and fit $\gamma = 0.44$ on masked diffusion models, so that the learning rate and the momentum stay constant for $B \propto T^{0.44}$.
[e] Fitted on the training runs released by Li et al. (2025a).
[f] Stated in the number of iterations T/B for a static batch size. The authors also derive a batch-size schedule within a run.
[g] Scaling laws derived for discrete diffusion language models. The learning-rate exponent combines $B \propto T^{0.8225}$ with $\eta \propto B^{0.3412}$.

(Choshen et al., 2025; McLeish et al., 2025). Related work connects batch size to gradient noise and optimization efficiency (Shallue et al., 2019; McCandlish et al., 2018).

Hyperparameter prescriptions exhibit similar variation (see Table 1). With batch size fixed, Bjorck et al. (2025) find that the optimal AdamW learning rate decreases as $T^{-0.32}$, whereas joint tuning of batch size and learning rate yields an increase of $T^{0.31}$ (Li et al., 2025a). Reported batch-size exponents likewise range from 0.38 to 0.69. These differences may reflect experimental protocols, search grids, and hardware constraints such as rounding batch sizes to powers of two.

How each study scales the model changes the reading of its exponents: the learning-rate rule of Li et al. (2025a) scales as $d_{\text{ne}}^{-0.71}T^{0.31}$ and becomes about $T^{-0.40}$ when $d_{\text{ne}} \propto T$, and Bi et al. (2024) fit along a compute-optimal path on which model and budget grow together (note (a) of Table 1). The smaller batch exponents of Bergsma et al. (2025) and Ajroldi et al. (2026) need not contradict our analysis: below the BST threshold, the leading stochastic term is batch-size-independent after tuning η and α, and BST selects the largest such batch.

Model Scaling Parameterizations. The maximal-update parameterization preserves stable, non-vanishing feature updates at infinite width (Yang & Hu, 2021) and enables hyperparameter transfer across widths (Yang et al., 2021). Such parameterizations are architecture-dependent: structured state-space models as well as Mixture of Experts models require their own scaling rules (Vankadara et al., 2024; 2026). Extensions address depth (Dey et al., 2025) and training duration (Mlodozeniec et al., 2026). The latter uses stochastic differential-equation approximations, which also yield batch-size transfer rules for adaptive optimizers (Malladi et al., 2022; Compagnoni et al., 2025).

Scaling Rules from Convergence Bounds. Several works derive hyperparameter prescriptions directly from optimization guarantees. Li et al. (2026) analyze convex SGD to motivate learning rate and batch schedules. Shulgin et al. (2026) instead optimize a stationarity-bound proxy for methods based on normalized linear minimization oracles. These guarantees control the gradient norm, which decays as $T^{-1/4}$ under smoothness and bounded variance alone (Arjevani et al., 2023),

and converting it through μ-KŁ would give a loss gap of the same order. Using the condition inside the analysis gives a loss gap of order $T^{-1/3}$, matched by our lower bound.

Closest to our work, Islamov et al. (2026) derive BST rules for stochastic conditional-gradient methods (SGG) under μ-KŁ in a general norm. Because their unit-ball linear minimization oracle discards the magnitude of the momentum vector, their guarantee requires a Frank-Wolfe step size to be proportional to $T^{-1/3}$. By contrast, the update magnitudes of SGDM and AdamW can decrease with the momentum magnitude while the learning rate remains constant.

Gradient Domination and Lower Bounds. Gradient-domination conditions trace back to Polyak (1963); Łojasiewicz (1963); Kurdyka (1998) and support convergence analyses for a broad class of optimization methods (Attouch et al., 2013; Bolte et al., 2013; Frankel et al., 2015). Local Łojasiewicz inequalities for analytic and subanalytic functions further motivate the μ-KŁ condition near a minimizer, with domination expressed through the loss gap (Bochnak & Łojasiewicz, 1971; Loi, 2016). We provide more concrete μ-KŁ guarantees for neural networks in Appendix F.

Existing gradient-domination analyses of Adam consider deterministic gradients (Barakat & Bianchi, 2020; Vaswani et al., 2026). We analyze stochastic gradients with decoupled weight decay. Masiha et al. (2026b;a) establish deterministic and stochastic lower bounds under gradient domination. In Section G.6, we compare their constructions with ours in detail.

3 Problem Formulation, Notation, and Assumptions

Notation. We write $x \in \mathbb{R}^d$ for the parameters of the model and equip $\mathbb{R}^d$ with the inner product $\langle \cdot, \cdot \rangle$ and its induced Euclidean norm $\| \cdot \|_2$. Our goal is to minimize the training loss f, which is bounded below by $f^\star := \inf_{x \in \mathbb{R}^d} f(x) > -\infty$. We assume access to a stochastic first-order oracle that returns unbiased gradient estimates $g(x, \xi)$, and we abbreviate $g_k := g(x_k, \xi_k)$. With a total budget of T tokens and batches of B tokens, an optimizer performs $K = T/B$ updates, and we consider SGD with momentum (SGDM) and AdamW, whose updates are given in Algorithms 1 and 2. The symbols $\cong$, $\gtrsim$, and $\lesssim$ denote equality and inequalities up to numerical and logarithmic factors.

Assumptions. Throughout, $\mathcal{X} \subseteq \mathbb{R}^d$ denotes a compact set on which the following assumptions hold. We assume that $\mathcal{X}$ contains the convex hull of the iterates of SGDM and AdamW.

Assumption 1. *The function f is L-smooth in $\mathbb{R}^d$: $\|\nabla f(x) - \nabla f(y)\|_2 \le L\|x - y\|_2, \forall x, y \in \mathbb{R}^d$.*

Assumption 2. *The function f satisfies the μ-KŁ condition for some $\mu > 0$ in $\mathcal{X}$, namely,*

$$\|\nabla f(x)\|_2 \ge \mu(f(x) - f^\star), \quad \textit{for all } x \in \mathcal{X}.$$

Assumption 3. *We have access to the unbiased stochastic oracle $g(x, \xi)$ of the gradient $\nabla f(x)$ with σ^2-bounded variance, namely, for all $x \in \mathcal{X}$, we have*

$$\mathbb{E}_\xi[g(x, \xi)] = \nabla f(x), \quad \mathbb{E}_\xi[\|g(x, \xi) - \nabla f(x)\|_2^2] \le \sigma^2.$$

Moreover, the gradient noise variance scales as $\sigma^2 = \sigma_\star^2/B$, where B is the batch size.

Assumption 4. *There exists a constant $G \ge 0$ such that for all coordinates i and iterations k of AdamW, we have $|g_k^i| \le G$.*

We use Assumption 4 only in the AdamW analysis, as is standard for adaptive methods (Reddi et al., 2018; Zaheer et al., 2018).

3.1 Scaling Problem and Description of Our Approach

Consider a model with training loss f satisfying Assumptions 1–3 with model constants $c = (L, \mu, \sigma_\star)$, and let $\mathcal{A}$ be an optimizer with hyperparameters θ (e.g., batch size, learning rate, weight decay). We write $x_{\mathcal{A}}(\theta)$ for the output of $\mathcal{A}$ after processing a token horizon T, where the dependence is implicit. Succinctly, our goal then is to find the optimal hyperparameters of $\mathcal{A}$ by solving

$$\theta^\star_{\mathcal{A}}(T, c) := \arg\min_\theta \ \mathbb{E}_\xi[f(x_{\mathcal{A}}(\theta)) - f^\star]. \tag{1}$$

For a small model at an economical token budget T_0, hyperparameter sweeps can accurately locate $\hat{\theta}_{\mathcal{A}}(T_0, c_0) \approx \theta^\star_{\mathcal{A}}(T_0, c_0)$. At the larger horizon T_1, however, such sweeps are computationally

expensive and not sustainable. We therefore turn to first principles to determine how $\theta^\star_{\mathcal{A}}(T, c)$ should scale with T and c. In the sequel, Theorems 1 and 2 give convergence bounds for `SGDM` and `AdamW` (e.g., $\varepsilon_{\texttt{AdamW}}(T, c, \theta)$) that upper-bound $\min_\theta \mathbb{E}[f(x_{\mathcal{A}}(\theta)) - f^\star] \le \varepsilon_{\mathcal{A}}(T, c, \theta)$. Using θ optimized by our analysis makes the resulting loss closer to the true minimum:

$$\min_\theta \mathbb{E}_\xi \left[f(x_{\mathcal{A}}(\theta)) - f^\star\right] \le \varepsilon_{\mathcal{A}}(T, c, \theta), \tag{2}$$

where $\varepsilon_{\mathcal{A}}(T, c, \theta)$ is defined in (3) and (5) for `SGDM` and `AdamW`, respectively. Making the proxy on the right-hand side of (2) as close as possible to the true solution leads to the BST scaling rule.

Definition 3.1 (BST scaling rule)**.** *We define the BST scaling rule as the hyperparameter choice $\theta^{\mathrm{BST}}_{\mathcal{A}}$ that minimizes $\varepsilon_{\mathcal{A}}(T, c, \theta)$ in θ subject to using the largest feasible batch size and weight decay*[1]*. For each hyperparameter in θ, the optimized choices take the form*

$$\theta^{\mathrm{BST}}_{\mathcal{A}} = \phi(c) T^\kappa d^\omega,$$

where the ϕ is some function of constants c, and κ and ω may differ across hyperparameters, and the dependence on model size d enters through the norm equivalence used in the analysis to guarantee convergence. Under this choice, the resulting suboptimality (ignoring logarithmic factors) satisfies

$$\varepsilon_{\mathcal{A}}(T, c, \theta^{\mathrm{BST}}_{\mathcal{A}}) = \mathcal{M}(c)\, T^{-1/3}$$

for some $\mathcal{M}$ that depends only on the model constants c. As compared to the standard scaling laws convention, note that by our definition, $\varepsilon_{\mathcal{A}}$ (i.e., suboptimality) is already debiased by $f^\star$.

Scaling procedure. We then use the BST rule to transfer hyperparameters $\hat{\theta}_{\mathcal{A}}(T_0, c_0)$, tuned at the small horizon T_0, to the target horizon T_1 with constants c_1: each hyperparameter is rescaled independently by the ratio of the corresponding factors in $\theta^{\mathrm{BST}}_{\mathcal{A}}$ evaluated at (T_1, c_1) versus (T_0, c_0),

$$\hat{\theta}_{\mathcal{A}}(T_1, c_1) = \hat{\theta}_{\mathcal{A}}(T_0, c_0) \cdot \frac{\theta^{\mathrm{BST}}_{\mathcal{A}}(T_1, c_1)}{\theta^{\mathrm{BST}}_{\mathcal{A}}(T_0, c_0)} = \hat{\theta}_{\mathcal{A}}(T_0, c_0) \cdot \frac{\phi(c_1)}{\phi(c_0)} \cdot \frac{T_1^\kappa}{T_0^\kappa} \cdot \frac{d_1^\omega}{d_0^\omega},$$

for each hyperparameter.

3.2 Transfer in Practice

The derived BST scaling rules depend on problem constants such as μ and L through the function ϕ. Islamov et al. (2026) handle this by estimating these constants during training and fitting power laws in model depth and width. In this approach, we need to choose the norm in which the constants are measured. This choice is natural for `Scion`, which can be viewed as descent in operator norms (Pethick et al., 2025), but no such canonical norm exists for `AdamW`. The fitted constants are also highly sensitive to measurement accuracy, which is hard to guarantee at scale.

We instead sidestep estimating c: following established practice, we co-scale width and depth in CompleteP (Dey et al., 2025), which preserves stable, predictable training as the model grows. We then apply the BST scaling exponents only to adjust hyperparameters with the token budget T and model size d, ignoring any change in the constants themselves (i.e., we set $\phi(c_1)/\phi(c_0) = 1$). We expect that CompleteP keeps this change in problem constants negligible across scales, though a precise characterization of how c affects the BST scaling rules is left to future work.

3.3 Main Challenges of Our Approach

The role of μ-KŁ condition. We follow Islamov et al. (2026), who show that Assumption 2 is a good approximation in practice: the gradient norm and training loss follow a linear fit once the loss drops below some threshold. As additional evidence for using the μ-KŁ condition in the convergence analysis, we analyze gradient flow for a two-layer linear network and prove that μ vanishes at the initial saddle and recovers uniformly after the saddle is escaped (see Section F.1). Appendix F gives explicit constants μ in three settings: two-layer linear networks under gradient flow, a bounded

[1]This choice is critical in increasing model flop utilization (MFU) in training (Chowdhery et al., 2023; Enskat & Wiesner, 2026); see Section C.4. We choose the largest possible weight decay to improve the performance and stability in long-horizon training.

readout after a normalization layer under sufficiently rich features, and the NTK regime near initialization under spectral assumptions on the data.

Why not PŁ? Another widely used structural assumption is the Polyak–Łojasiewicz (PŁ) condition (Polyak, 1963; Karimi et al., 2016), which in theory typically requires over-parameterization, i.e., more parameters than data points (Liu et al., 2020). LLM pre-training, however, is usually conducted in the under-parameterized regime, with more tokens than parameters (Hoffmann et al., 2022a). In Example 7, we show that for a two-layer linear network, the μ-KŁ ratio stays above $2\sqrt{2}$ on the low-loss set, while the PŁ ratio goes to zero. This further supports our choice of μ-KŁ over PŁ.

Tightness of the analysis. To obtain a tight approximation gap in Equation (2), we provide lower bounds for *zero-respecting* algorithms (see Definition G.4), which show that our upper bounds have the right order in L, σ, and the convergence rate in ε.

4 Convergence Analysis and BST Scaling Rule

This section presents convergence guarantees for SGDM and AdamW, together with the BST scaling rules that follow from these bounds.

4.1 Convergence Analysis of SGD with Momentum

We start with SGD with momentum (SGDM), whose update is

$$x_{k+1} = x_k - \eta m_k, \quad m_k = (1-\alpha)m_{k-1} + \alpha g_k.$$

Theorem 1. *Let Assumptions 1, 2, 3 hold, let $\varepsilon > 0$, and set the hyperparameters of SGDM (Alg. 1) such that*

$$\alpha \cong \min\left\{1, \left(\tfrac{\varepsilon\mu}{\sigma}\right)^2\right\}, \quad \eta \cong \min\left\{\tfrac{1}{L}, \tfrac{(\varepsilon\mu)^2}{L\sigma^2}, \tfrac{1}{\mu^2\varepsilon}, \tfrac{\varepsilon}{\sigma^2}\right\}, \quad K \cong \max\left\{1, \tfrac{L}{\mu^2\varepsilon}, \tfrac{L\sigma^2}{\mu^4\varepsilon^3}, \tfrac{\sigma^2}{\varepsilon^2\mu^2}\right\}.$$

Then, $\mathbb{E}[f(x_K) - f^\star] \le \varepsilon$.

The complete proof is deferred to Appendix D. Under a fixed token budget T, this convergence rate implies the following error floor.

Corollary 1. *Under the setup of Theorem 1, SGDM achieves the optimization error*

$$\varepsilon_{\texttt{SGDM}}(T, c) \cong \max\left\{\tfrac{LB}{\mu^2 T}, \left(\tfrac{L\sigma_\star^2}{\mu^4 T}\right)^{1/3}, \left(\tfrac{\sigma_\star^2}{\mu^2 T}\right)^{1/2}\right\}, \tag{3}$$

after T/B iterations. If $T \gtrsim (\mu\sigma_\star/L)^2$, the BST scaling rule for SGDM is $B \propto T^{2/3}$ with constant η and α, and the error floor is $\varepsilon \propto T^{-1/3}$.

In contrast to the error floor of SCG (Islamov et al., 2026), small batch sizes do not hurt the error floor of SGDM. The error grows once the batch size becomes large enough for the first term in (3) to dominate. Intriguingly, the error floor of SGDM has two batch-size-independent terms: $\varepsilon \propto T^{-1/2}$ in the small-T regime, and $\varepsilon \propto T^{-1/3}$ in the large-T regime. The full proof is deferred to Appendix D.

4.2 Convergence Analysis of AdamW

Next, we switch to the analysis of AdamW (Loshchilov & Hutter, 2019), a common optimizer in LLM pre-training. Its update can be written as

$$m_k = (1-\alpha)m_{k-1} + \alpha g_k, \quad v_k = (1-\beta)v_{k-1} + \beta(g_k \odot g_k), \quad x_{k+1} = (1-\eta\lambda)x_k - \eta\frac{m_k}{\sqrt{v_k} + \nu},$$

where ν is the stability constant. The next theorem gives sufficient conditions for convergence.

Theorem 2 (Simplified)**.** *Let Assumptions 1, 2, 3, and 4 hold, let $0 < \varepsilon < 1$ and set the hyperparameters of AdamW (Alg. 2) such that*

$$\alpha \cong \min\left\{1, \tfrac{Bp_l\mu^2\varepsilon^2}{p_u\sigma_\star^2}\right\}, \quad \eta \cong \tfrac{p_l^{1/2}\alpha}{p_u^{3/2}(L+\mu^2)}, \tag{4}$$

$$K \cong \tfrac{1}{\eta p_l \mu^2 \varepsilon}, \quad \lambda \le \min\left\{\tfrac{1}{2\eta}, \tfrac{p_l\mu\varepsilon}{\sqrt{d}\left(\|x_0\|_\infty + \sqrt{p_u p_l^{-1}\mu^{-2}}\,(1+\log_2(T/B))\right)}\right\},$$

where $p_l \coloneqq 1/(G+\nu)$ and $p_u \coloneqq 1/\nu$. Then, $\mathbb{E}[f(x_K) - f^\star] \le \varepsilon$.

The full statement and proof are provided in Appendix E. Our theoretical analysis assumes a constant learning rate; we leave the study of the effect of learning-rate schedules on the convergence analysis to future work. Under a fixed token budget, Theorem 2 gives the following error floor.

Corollary 2. *Under the setup of Theorem 2,* `AdamW` *achieves the optimization error*

$$\varepsilon_{\texttt{AdamW}}(T,c) \cong \max\left\{\frac{p_u^{3/2}(L+\mu^2)B}{p_l^{3/2}\mu^2 T}, \left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4 T}\right)^{1/3}\right\}. \tag{5}$$

after T/B *iterations. For* T *above the threshold* (37)*, the BST scaling rule for* `AdamW` *is* $B \propto T^{2/3}$ *with constant* η*,* α *and a sufficient weight decay* $\lambda \propto 1/(\sqrt{d}\,T^{1/3})$*, and the error floor is* $\varepsilon \propto T^{-1/3}$.

`AdamW` inherits the `SGDM` rule: the same $T^{-1/3}$ floor and batch exponent, and a constant learning rate and momentum, with the preconditioner entering only through p_u/p_l. Islamov et al. (2026) obtain the same rate for `SCG` method. The one new element is the weight decay, whose sufficient value decreases with the optimization problem dimension and the token budget.

5 Lower Bounds under μ-KŁ Condition

Based on Assumptions 1, 2, and 3, we now establish a matching lower bound, showing that the dependence of our upper bounds on the target accuracy, the smoothness constant, and the noise level are tight for zero-respecting algorithms. As shown by Masiha et al. (2026a), the global versions of the μ-KŁ condition and L-smoothness are incompatible in the setting considered here. We therefore restrict the optimization domain to the box $\mathcal{X} := [0,2]^d$ and construct a hard instance satisfying all three assumptions on $\mathcal{X}$.[2] The following theorem gives the precise lower-bound statement.

Theorem 3 (Simplified)**.** *For every* $L > 0$*,* $\sigma \geq 0$*, and sufficiently small* $\varepsilon > 0$*, there exist a dimension* $d \geq 2$*, a function* $f : \mathbb{R}^d \to \mathbb{R}$*, and a stochastic first-order oracle such that Assumptions 1, 2 and 3 hold with* $\mu = 1$ *on* $\mathcal{X} = [0,2]^d$*. Moreover, any zero-respecting stochastic algorithm whose oracle queries and output* $\widehat{x}_K$ *belong to* $\mathcal{X}$ *satisfies* $\mathbb{E}[f(\widehat{x}_K) - f^\star] \leq \varepsilon$ *must use at least*

$$K = \Omega\left(L/\varepsilon + L\sigma^2/\varepsilon^3\right) \quad \textit{iterations.} \tag{6}$$

This lower bound shows that, within the class of zero-respecting stochastic algorithms considered above, neither the deterministic term L/ε nor the stochastic term $L\sigma^2/\varepsilon^3$ in the upper bounds can be improved in order. When the variance of a minibatch gradient scales as $\sigma^2 = \sigma_\star^2/B$, balancing these two contributions yields the same batch-size scaling as the BST rule derived from our upper bounds.

6 Experiments

The experimental section evaluates the predictive power of the proposed BST scaling rules. We first test transfer across token horizons using NanoGPT models (Jordan et al., 2024), and then study transfer across both token horizons and model sizes using Peri-LN models (Kim et al., 2025). Detailed experimental settings are provided in Appendices B and C, respectively.

6.1 Tuning of the Base Model with AdamW

We first tune a base configuration for a 124M-parameter NanoGPT model at $T = 1.3$B tokens, which we later transfer to larger token budgets. The three-stage tuning procedure is described in detail in Section B.1; here we summarize the main conclusions.

Batch size and learning rate. We start with sweeping batch size B, learning rate η, and multiplicative learning-rate factor `lr_ratio` between the embedding/head parameters and the hidden-layer parameters, while holding the remaining hyperparameters fixed to default values. The best-performing configuration within the tested grid is $B = 128, \eta = 0.00125,$ `lr_ratio` $= 16$; see sweeps in Figures B.1 and B.2.

[2] While preparing this manuscript, we became aware of the work of Masiha et al. (2026a), which also establishes lower bounds for the class of μ-KŁ functions. We provide a detailed comparison in Section G.6.

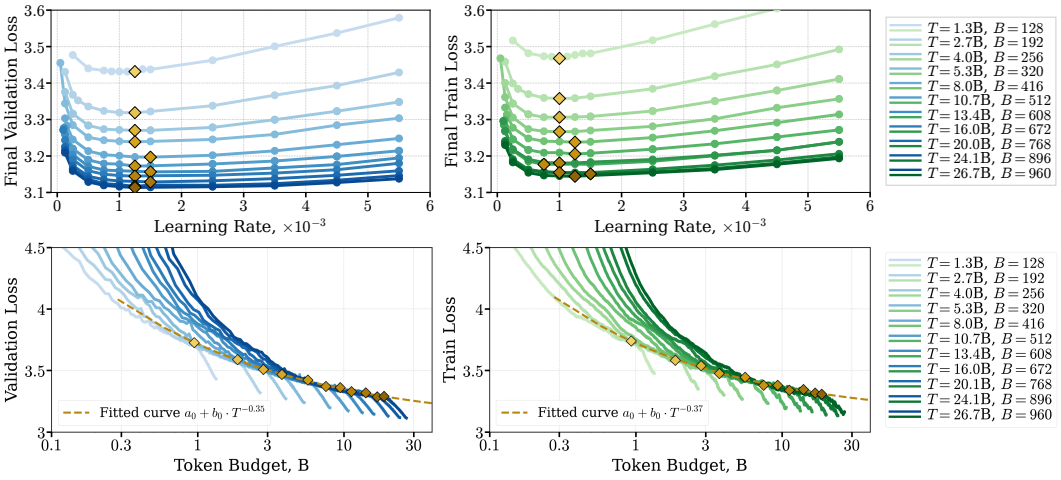


Figure 2: **Top: learning rate transfers under BST scaling law. Bottom: both validation and training losses decay roughly as $T^{-1/3}$, supported by theory; see Corollary 2.** The final training loss of `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset when varying the token budget T and adjusting the batch size according to the BST rule $B \propto T^{2/3}$. In the top row, a rhombus highlights the best validation and training losses; in the bottom row, the power law is fitted at the last 100 iterations before the cooldown phase. Training loss is averaged over 500 steps.

Weight decay. We next fix $B = 128$, $\eta = 0.00125$, and `lr_ratio` $= 16$, and sweep the weight-decay λ, keeping the remaining hyperparameters unchanged. The validation and training losses are minimized near $\lambda = 0.1$, consistent with common language-model training recipes (Li et al., 2025a; Touvron et al., 2023; Bi et al., 2024); see the sweeps in Figure B.3.

Momentum parameters. Finally, we grid-search the momentum parameters (α, β), keeping all other hyperparameters at their tuned or default values. The best choice, $(\alpha, \beta) = (0.1, 0.05)$. It corresponds to the default choice $(\beta_1, \beta_2) = (0.9, 0.95)$, widely used in language-model pre-training (Mehta et al., 2024; Touvron et al., 2023; Bi et al., 2024) (see full sweep in Figure B.4). Unless stated otherwise, all subsequent experiments use these tuned hyperparameters.

6.2 BST Rule Across Token Horizons

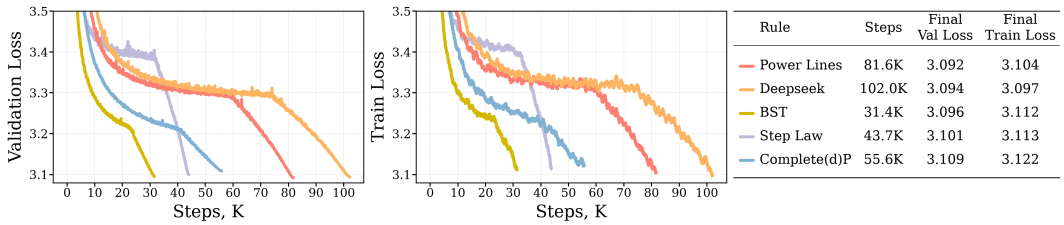
We now test how well hyperparameters transfer across token budgets. Starting from the `AdamW` configuration tuned at $T = 1.3$B tokens, we increase T while scaling B and λ per the BST rule: $B \propto T^{2/3}$, $\lambda \propto T^{-1/3}$ (model size fixed), and keeping all other hyperparameters fixed. For each (T, B), we sweep the learning rate η to check whether its optimum stays invariant under this scaling; results are reported in Figure 2, while the exact values of hyperparameters are provided in Table B.2.

We find that the empirically optimal learning rate transfers well across token horizons. In terms of training loss, it consistently lies within $\eta \in [0.75, 1.5] \times 10^{-3}$, while for validation loss it stays within $\eta \in [1.25, 1.5] \times 10^{-3}$. This result supports the predictive power of the theoretically derived BST rule: when the token budget changes for a fixed model, scaling the batch size as $B \propto T^{2/3}$ approximately preserves the optimal learning rate.

We next check whether the error-floor behavior predicted by Theorem 2 appears empirically, by fitting the power law $a_0 + b_0 T^{-c_0}$ ($a_0, b_0, c_0 \geq 0$) to training and validation losses measured at 70% of training, just before cooldown. Following Schaipp (2026), we estimate parameters by minimizing the Huber loss with threshold $\delta = 10^{-3}$. Figure 2 shows that training loss decreases approximately as $T^{-1/3}$ across token budgets, matching our theoretical rate. The validation loss behaves the same way, which we conjecture holds whenever the model is under-parameterized, as in LLM pre-training.

6.3 Comparison of Scaling Laws

Starting from the tuned `AdamW` configuration of the base model at $T = 1.3$B tokens, we compare several hyperparameter-scaling rules at $T = 40.1$B. Specifically, we evaluate BST against DeepSeek (Bi et al., 2024), Step Law (Li et al., 2025a), Power Lines (Bergsma et al., 2025), and Complete(d)P



Rule	Steps	Final Val Loss	Final Train Loss
Power Lines	81.6K	3.092	3.104
Deepseek	102.0K	3.094	3.097
BST	31.4K	3.096	3.112
Step Law	43.7K	3.101	3.113
Complete(d)P	55.6K	3.109	3.122

Figure 3: **Principled BST scaling law requires least steps to perform competitively with empirically fitted ones in the long-horizon training for a fixed model.** The comparison of scaling laws for `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset at the token budget $T = 40.1$B. BST requires the fewest steps to perform competitively. Note that differences to the best-performing scaling laws are within the noise level (see Section C.3).

(Mlodozeniec et al., 2026). For Complete(d)P we use the batch-size scaling $B \propto T^{0.44}$ following (Bethune et al., 2026). Hyperparameters not covered by a given rule are kept at their tuned base values, while all remaining hyperparameters are transferred as prescribed: see Table B.2 for more details. As shown in Figure 3, BST and Power Lines attain the lowest loss at $T = 40.1$B, with BST using a larger batch size, which is often preferable for computational efficiency (Narayanan et al., 2021). This result demonstrates the effectiveness of the BST scaling law against other empirically derived scaling laws. Further implementation details and comparisons are provided in Section B.2.

6.4 Co-scaling of Model Size and Token Budget

The scaling laws summarized in Table 1 use different measures of model scale: some depend on the total parameter count d, others on the non-embedding parameter count d_{ne}, while Bi et al. (2024) use non-embedding FLOPs per token. To compare these rules consistently across model sizes, we co-scale model width and depth using CompleteP (Dey et al., 2025). In this section, we switch to the Peri-LN Transformer model to improve training stability and hyperparameter transfer across scales.

Similar to Section 6.1, we tune `AdamW`'s hyperparameters for the base 162M model at $T = 1.62$B tokens (see details in Section C.1). We then build a ladder of four Peri-LN models with 162M, 304M, 834M and 1.4B parameters, and aspect ratio 64. Each model is trained for $T = 20$B tokens and Table C.2 shows the final values for each of the scaling laws at every model size. Transferring hyperparameters tuned for the base model along this ladder assumes that the parameterization approximately preserves model-scale-invariant dynamics, which has been argued by Bordelon et al. (2024); Dey et al. (2025) and which we partially verify using coordinate checks in Section C.2.

The results are reported in Figure 1. With increasing model size, Power Lines is the only scaling law that recovers competitive performance within the noise floor compared to our BST scaling rule. However, Power Lines requires more steps, the most among the scaling laws that we compare. This gap would only grow with the number of steps T, and result in significantly more GPU hours to reach the same loss level.

7 Conclusion and Future Work

In this work, we derived token-budget-aware BST scaling laws for `AdamW` from optimization guarantees under the μ-KŁ condition. A matching lower bound in L, σ, ε establishes the tightness of both our convergence analysis and the resulting scaling exponents. BST prescribes how hyperparameters should vary with the token horizon and model size without relying on expensive empirical fits. Our numerical evidence sufficiently demonstrate predictive transfer across token budgets and model sizes, with performance competitive with empirically fitted rules.

Important directions for future work include incorporating learning-rate schedules into the analysis, characterizing how the problem constants vary with model configuration and training horizon, and testing whether the constants measured along training can better predict the transfer across model sizes. The latter may further improve BST when transferring to substantially larger models and allow for characterizing such scaling laws across different model families, such as MoEs, SSMs, and Looped Transformers.

REFERENCES

Niccoló Ajroldi, Diana Alexandra Onut, Haider Al-Tahan, J örg Franke, Sampo Pyysalo, Jenia Jitsev, and Aaron Klein. Deriving scaling laws for openeurollm models: Learning rate, batch size and loss. *arXiv preprint arXiv:2608.28308*, 2026.

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient Flows in Metric Spaces and in the Space of Probability Measures*. Birkhäuser, 2008.

Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 2023.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. In *International Conference on Learning Representations*, 2019.

Hédy Attouch, Jérôme Bolte, and Benar Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical Programming*, 2013.

Francis Bach. Closed-form dynamics beyond quadratics. Blog post, `https://francisbach.com/closed-form-dynamics/`, 2026. Accessed: 2026-09-12.

Anas Barakat and Pascal Bianchi. Convergence rates of a momentum algorithm with bounded adaptive step size for nonconvex optimization. In *Proceedings of The 12th Asian Conference on Machine Learning*, 2020.

Shane Bergsma, Nolan Simran Dey, Gurpreet Gosal, Gavia Gray, Daria Soboleva, and Joel Hestness. Power lines: Scaling laws for weight decay and batch size in LLM pre-training. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Louis Bethune, Victor Turrisi, Bruno Kacper Mlodozeniec, Pau Rodriguez Lopez, Lokesh Boominathan, Nikhil Bhendawade, Amitis Shidani, Joris Pelemans, Theo X Olausson, Devon Hjelm, et al. The design space of tri-modal masked diffusion models. *arXiv preprint arXiv:2602.21472*, 2026.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024.

Johan Bjorck, Alon Benhaim, Vishrav Chaudhary, Furu Wei, and Xia Song. Scaling optimal LR across token horizons. In *The Thirteenth International Conference on Learning Representations*, 2025.

J. Bochnak and S. Łojasiewicz. A converse of the kuiper-kuo theorem. In C.T.C. Wall (ed.), *Proceedings of Liverpool Singularities — Symposium I*, pp. 254–261, Berlin, Heidelberg, 1971. Springer Berlin Heidelberg. ISBN 978-3-540-36531-0.

Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 2013.

Blake Bordelon, Hamza Chaudhry, and Cengiz Pehlevan. Infinite limits of multi-head transformer dynamics. *Advances in Neural Information Processing Systems*, 37:35824–35878, 2024.

Lukas Braun, Clémentine Dominé, James Fitzgerald, and Andrew Saxe. Exact learning dynamics of deep linear networks with prior knowledge. In *Advances in Neural Information Processing Systems*, 2022.

Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points I. *Mathematical Programming*, 2019.

Lénaïc Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems*, 2018.

Youngmin Cho and Lawrence Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, 2009.

Leshem Choshen, Yang Zhang, and Jacob Andreas. A hitchhiker's guide to scaling law estimation. In *Forty-second International Conference on Machine Learning*, 2025.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 2023.

Enea Monzio Compagnoni, Tianlin Liu, Rustem Islamov, Frank Norbert Proske, Antonio Orvieto, and Aurelien Lucchi. Adaptive methods through the lens of SDEs: Theoretical insights on the role of noise. In *The Thirteenth International Conference on Learning Representations*, 2025.

Nolan Simran Dey, Bin Claire Zhang, Lorenzo Noci, Mufan Li, Blake Bordelon, Shane Bergsma, Cengiz Pehlevan, Boris Hanin, and Joel Hestness. Don't be lazy: Completep enables compute-efficient deep transformers. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Aleksandr Dremov, Alexander Hägele, Atli Kosson, and Martin Jaggi. Training dynamics of the cooldown stage in warmup-stable-decay learning rate scheduler. *Transactions on Machine Learning Research*, 2025.

Niklas Enskat and Philipp Wiesner. Evaluating mfu as a proxy for gpu power for energy-aware simulation of llm training. *arXiv preprint arXiv:2608.03880*, 2026.

Ilyas Fatkhullin, Jalal Etesami, Niao He, and Negar Kiyavash. Sharp analysis of stochastic optimization under global kurdyka-lojasiewicz inequality. In *Advances in Neural Information Processing Systems*, 2022.

Oleg Filatov, Jiangtao Wang, Jan Ebert, and Stefan Kesselheim. Optimal scaling needs optimal norm. In *High-dimensional Learning Dynamics 2026*, 2026.

Xavier Fontaine, Valentin De Bortoli, and Alain Durmus. Convergence rates and approximation results for sgd and its continuous-time counterpart. In *Proceedings of Thirty Fourth Conference on Learning Theory*, 2021.

Pierre Frankel, Guillaume Garrigos, and Juan Peypouquet. Splitting methods with variable metric for kurdyka–łojasiewicz functions and general convergence rates. *Journal of Optimization Theory and Applications*, 2015.

Kenji Fukumizu. Effect of batch learning in multilayer neural networks. In *Proceedings of the 5th International Conference on Neural Information Processing*, 1998.

Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, 2019.

Alexander Hägele, Elie Bakouch, Atli Kosson, Loubna Ben allal, Leandro Von Werra, and Martin Jaggi. Scaling laws and compute-optimal training beyond fixed training durations. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Moritz Hardt, Tengyu Ma, and Benjamin Recht. Gradient descent learns linear dynamical systems. *Journal of Machine Learning Research*, 2018.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, 2022a.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, 2022b.

Shengding Hu, Yuge Tu, Xu Han, Ganqu Cui, Chaoqun He, Weilin Zhao, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Xinrong Zhang, Zhen Leng Thai, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, dahai li, Zhiyuan Liu, and Maosong Sun. MiniCPM: Unveiling the potential of small language models with scalable training strategies. In *First Conference on Language Modeling*, 2024.

Rustem Islamov, Roman Machacek, Aurelien Lucchi, Tony Silveti-Falls, Eduard Gorbunov, and Volkan Cevher. On the role of batch size in stochastic conditional gradient methods. In *Forty-third International Conference on Machine Learning*, 2026.

Arthur Jacot, Franck Gabriel, and Clement Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, 2018.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

Keller Jordan, Jeremy Bernstein, Brian Rappazzo, Fernbear, Vlado, Jiacheng Yu, Francesco Cesista, Ben Koszarsky, and Grad62304977. moddednanogpt: Speedrunning the nanogpt baseline. `https://github.com/KellerJordan/modded-nanogpt`, 2024. Version 2024a.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Kedar Karhadkar, Michael Murray, and Guido Montufar. Bounds for the smallest eigenvalue of the NTK for arbitrary spherical data of arbitrary dimension. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2016.

Jeonghoon Kim, Byeongchan Lee, Cheonbok Park, Yeontaek Oh, Beomjun Kim, Taehwan Yoo, Seongjin Shin, Dongyoon Han, Jinwoo Shin, and Kang Min Yoo. Peri-LN: Revisiting normalization layer in the transformer architecture. In *Forty-second International Conference on Machine Learning*, 2025.

Daniel Kunin, Allan Raventos, Clémentine Carla Juliette Dominé, Feng Chen, David Klindt, Andrew M Saxe, and Surya Ganguli. Get rich quick: exact solutions reveal how unbalanced initializations promote rapid feature learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Krzysztof Kurdyka. On gradients of functions definable in o-minimal structures. *Annales de l'Institut Fourier*, 48(3):769–783, 1998.

Tim Large, Yang Liu, Minyoung Huh, Hyojin Bahng, Phillip Isola, and Jeremy Bernstein. Scalable optimization in the modular norm. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Houyi Li, Wenzhen Zheng, Qiufeng Wang, Hanshan Zhang, Zili Wang, Shijie Xuyang, Yuantao Fan, Zhenyu Ding, Haoying Wang, Ning Ding, et al. Predictable scale: Part i, step law–optimal hyperparameter scaling law in large language model pretraining. *arXiv preprint arXiv:2503.04715*, 2025a.

Huan Li, Yiming Dong, and Zhouchen Lin. On the $o(\frac{\sqrt{d}}{K^{1/4}})$ convergence rate of adamw measured by ℓ_1 norm. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b.

Jiaxiang Li, Zhiqi Bu, and Shiyun Xu. Towards joint scaling laws with optimal batch size schedules. *arXiv preprint arXiv:2607.27731*, 2026.

Chaoyue Liu, Libin Zhu, and Mikhail Belkin. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *Applied and Computational Harmonic Analysis*, 2020.

Chaoyue Liu, Libin Zhu, and Mikhail Belkin. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *Applied and Computational Harmonic Analysis*, 2022.

Ta Lê Loi. Łojasiewicz inequalities in o-minimal structures. *Manuscripta Mathematica*, 150(1): 59–72, 2016.

Stanisław Łojasiewicz. Une propriété topologique des sous-ensembles analytiques réels. In *Les Équations aux Dérivées Partielles*, pp. 87–89. Éditions du Centre National de la Recherche Scientifique, Paris, 1963.

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

Sadhika Malladi, Kaifeng Lyu, Abhishek Panigrahi, and Sanjeev Arora. On the SDEs and scaling rules for adaptive gradient algorithms. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Saeed Masiha, Saber Salehkaleybar, Niao He, Negar Kiyavash, and Patrick Thiran. Stochastic second-order methods improve best-known sample complexity of SGD for gradient-dominated functions. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Saeed Masiha, Negar Kiyavash, and Patrick Thiran. Sharp first-order lower bounds under sublevel α-polyak-lojasiewicz conditions. *arXiv preprint arXiv:2606.28278*, 2026a.

Saeed Masiha, Saber Salehkaleybar, Niao He, Negar Kiyavash, and Patrick Thiran. Optimal local convergence rates of stochastic first-order methods under local alpha-pl. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, 2026b.

Sam McCandlish, Jared Kaplan, Dario Amodei, and OpenAI Dota Team. An empirical model of large-batch training. *arXiv preprint arXiv:1812.06162*, 2018.

Sean Michael McLeish, John Kirchenbauer, David Yu Miller, Siddharth Singh, Abhinav Bhatele, Micah Goldblum, Ashwinee Panda, and Tom Goldstein. Gemstones: A model suite for multi-faceted scaling laws. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Sachin Mehta, Mohammad Hossein Sekhavat, Qingqing Cao, Maxwell Horton, Yanzi Jin, Chenfan Sun, Seyed Iman Mirzadeh, Mahyar Najibi, Dmitry Belenko, Peter Zatloukal, and Mohammad Rastegari. OpenELM: An efficient language model family with open training and inference framework. In *Workshop on Efficient Systems for Foundation Models II @ ICML2024*, 2024.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

Hancheng Min, Salma Tarmoun, René Vidal, and Enrique Mallada. Convergence and implicit bias of gradient flow on overparametrized linear networks. *arXiv preprint arXiv:2105.06351*, 2021.

Bruno Kacper Mlodozeniec, Pierre Ablin, Louis Béthune, Dan Busbridge, Michal Klein, Jason Ramapuram, and marco cuturi. Completed hyperparameter transfer across modules, width, depth, batch and duration. In *The Fourteenth International Conference on Learning Representations*, 2026.

Deepak Narayanan, Mohammad Shoeybi, Jared Casper, Patrick LeGresley, Mostofa Mostofi, Md Patwary, Vijay Korthikanti, Dmitry Vainbrand, Prethvi Kashinkunti, Woomin Jeon, Xiao Bernauer, and Bryan Catanzaro. Efficient large-scale language model training on gpu clusters using megatron-lm. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. Association for Computing Machinery, 2021.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, 2004.

Quynh Nguyen, Marco Mondelli, and Guido F Montufar. Tight bounds on the smallest eigenvalue of the neural tangent kernel for deep relu networks. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, 2019.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In *NeurIPS*, 2024.

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

Boris T. Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 1963.

Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *The Fourteenth International Conference on Learning Representations*, 2014.

Fabian Schaipp. How to allocate your tokens? scaling laws with training steps and batch size. *arXiv preprint arXiv:2607.01487*, 2026.

Christopher J. Shallue, Jaehoon Lee, Joseph Antognini, Jascha Sohl-Dickstein, Roy Frostig, and George E. Dahl. Measuring the effects of data parallelism on neural network training. *Journal of Machine Learning Research*, 2019.

Egor Shulgin, Dimitri von Rütte, Tianyue H Zhang, Niccolò Ajroldi, Bernhard Schölkopf, and Antonio Orvieto. Deriving hyperparameter scaling laws via modern optimization theory. *arXiv preprint arXiv:2603.15958*, 2026.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

Salma Tarmoun, Guilherme Franca, Benjamin D Haeffele, and Rene Vidal. Understanding the dynamics of gradient flow in overparameterized linear models. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Berk Tinaz, Changzhi Xie, and Mahdi Soltanolkotabi. When both layers learn: Training dynamics of representing linear models via relu networks. *arXiv preprint arXiv:2606.04476*, 2026.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Leena Chennuru Vankadara, Jin Xu, Moritz Haas, and Volkan Cevher. On feature learning in structured state space models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Leena Chennuru Vankadara, Moritz Haas, Luke Hayward, Sebastian Bordt, and Alessandro Breccia. How to scale mixture-of-experts: From mup to the maximally scale-stable parameterization. *arXiv preprint arXiv:2605.14200*, 2026.

Sharan Vaswani, Yifan Sun, and Reza Babanezhad Harikandeh. Convergence of steepest descent and adam under non-uniform smoothness. In *Forty-third International Conference on Machine Learning*, 2026.

Roman Vershynin. High-dimensional probability. *Cambridge Series in Statistical and Probabilistic Mathematics*, 47, 2019.

Dimitri von Rütte, Janis Fluri, Omead Pooladzandi, Bernhard Schölkopf, Thomas Hofmann, and Antonio Orvieto. Scaling behavior of discrete diffusion language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Shuo Xie and Zhiyuan Li. Implicit bias of adamw: ℓ_∞-norm constrained optimization. In *Forty-first International Conference on Machine Learning*, 2024.

Greg Yang and Edward J. Hu. Tensor programs iv: Feature learning in infinite-width neural networks. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Greg Yang, Edward Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tuning large neural networks via zero-shot hyperparameter transfer. In *Advances in Neural Information Processing Systems*, 2021.

Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. In *Advances in Neural Information Processing Systems*, 2018.

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

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep ReLU networks. *arXiv preprint arXiv:1811.08888*, 2018.

Appendix

CONTENTS

A Usage of LLMs **17**

B NanoGPT Experimental Setup Description **17**

B.1 Tuning of the Base NanoGPT 124M Model with AdamW 17

B.2 Additional Details on the Comparison of Scaling Laws 19

C Additional Experiments and Details for Peri-LN model **21**

C.1 Tuning of the Base 162M Model with AdamW 21

C.2 Coordinate Checks and Problem Constants for the Co-scaled Ladder 23

C.3 Loss Dynamics Along the Ladder 24

C.4 MFU and Throughput 25

D Convergence Analysis of SGD with Momentum **26**

D.1 Auxiliary Lemmas 26

D.2 Proof of the Main Theorem 29

D.3 Error Floor and BST Rule 30

E Convergence Analysis for AdamW **32**

E.1 Auxiliary Lemmas 32

E.2 Proof of the Main Theorem 40

E.3 Error Floor and BST Scaling Rule for AdamW 44

F μ-KŁ Guarantees for Neural Networks **46**

F.1 Two-Layer Linear Networks 47

F.2 Analysis for a Linear Readout 50

F.3 A Two-Layer Mean-Field Perspective 53

F.4 The Lazy/NTK Regime 54

G Lower Bounds for Stochastic Methods under μ-KŁ **58**

G.1 Definition of Zero-chains 58

G.2 Scalar Building Blocks 58

G.3 The Unbiased Stochastic Masking Oracle 67

G.4 Zero-respecting Stochastic Methods 68

G.5 Lower Bound for Function Suboptimality 69

G.6 Detailed Comparison of Lower bounds in this Work and Masiha et al. (2026a) . . . 70

A USAGE OF LLMS

In this paper, we use large language models in the following way:

- To polish the writing, correct grammar and spelling.
- To execute and polish code.
- To help with the convergence analysis of `AdamW`, correct mistakes and typos in the derivations.
- To help proof-read and extend the μ-KŁ guarantees in Appendix F, correct mistakes and typos in the derivations.

B NANOGPT EXPERIMENTAL SETUP DESCRIPTION

Our NanoGPT experiments build on the codebase of Jordan et al. (2024). The implementation ties the input embedding and the output head, uses the scaled ReLU^2 activation from Large et al. (2024), replaces positional embeddings with rotary embeddings (Su et al., 2021), and uses RMSNorm (Zhang & Sennrich, 2019) instead of LayerNorm, together with QK-norm. Following Pethick et al. (2025); Islamov et al. (2026), all normalization layers have no learnable gains. Instead of cosine annealing, we use the WSD learning rate schedule (Hu et al., 2024). All experiments are run on the FineWeb dataset (Penedo et al., 2024) with the PyTorch implementation of `AdamW`.

We conduct all experiments in PyTorch (Paszke et al., 2019), including an implementation of `AdamW`. The training of the base NanoGPT model is conducted on a single A100 GPU.

Table B.1: The model configurations and training details of NanoGPT models with `AdamW`.

Hyperparameter	124M Model
Layers	12
Heads	6
Embedding Size	768
Sequence Length	1024
Weight Tying	Yes
Activation Function	ReLU^2
Vocabulary Size	50304
Dataset	FineWeb
Warmup	10% of the total token budget
Cooldown	28% of the total token budget
Cooldown shape	Linear Decay
Stepsize Schedule	WSD
Gradient Clipping	No
Momentum Parameters	$\alpha = 0.1, \beta = 0.05$, $(\beta_1 = 0.9, \beta_2 = 0.95)$ *unless stated otherwise*
Weight Decay	$\lambda = 0.1$ *unless stated otherwise*
`lr_ratio`	16
Precision	Mixed precision bf16
Device Batch Size	32
GPU Type	A100-40G
Implementation	`https://anonymous.4open.science/r/NanoGPT-462B/README.md`

B.1 TUNING OF THE BASE NANOGPT 124M MODEL WITH ADAMW

Our objective here is to identify a well-tuned base configuration from which hyperparameters can subsequently be transferred to larger token budgets and model sizes. We use a 124M-parameter NanoGPT model with 12 layers, 6 attention heads, a hidden dimension of 768, and tied embedding/head layers, following prior work (Pethick et al., 2025; Islamov et al., 2026). All base-model tuning runs use a budget of $T = 1.3 \times 10^9$ tokens, corresponding to approximately 10.5 tokens per parameter. Finally, our momentum notation (α, β) is related to the conventional AdamW notation by $\beta_1 = 1 - \alpha$ and $\beta_2 = 1 - \beta$.

Batch Size and Learning Rate Tuning. First, for the base model, we independently tune the batch size

$$B \in \{64, 128, 256, 512, 1024, 2048\},$$
$$\texttt{lr_ratio} \in \{1, 4, 16, 32\},$$
$$\eta \in \{0.25, 0.5, 1.5, 2.5, 3.5, 4.5, 5.5\} \times 10^{-4},$$

where hidden layers are trained with the learning rate η, while the embedding/head layer is trained with $\texttt{lr_ratio}\times\eta$. We use a weight decay parameter $\lambda = 0.1$ for the hidden layers, and $\lambda/\texttt{lr_ratio}$ for the embedding/head layer, so that the contraction factor $1 - \eta\lambda$ is the same across all layers. We keep $\alpha = 0.1, \beta = 0.05, \nu = 10^{-8}$, and $\lambda = 0.1$ fixed.

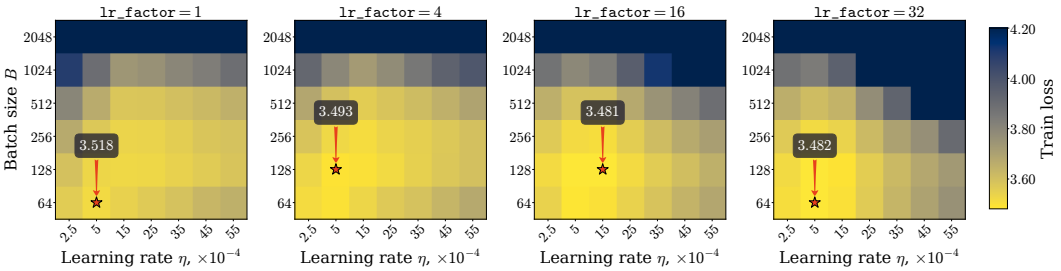


Figure B.1: The final training loss of `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset when varying the batch size, learning rate, and `lr_ratio`, while keeping $\alpha = 0.1, \beta = 0.05, \nu = 10^{-8}, \lambda = 0.1$ fixed. Train loss is averaged over 500 iterations.

In Figure B.1, we report the performance of `AdamW` when three hyperparameters (B, η, λ) are swept independently. We observe that the best performance of `AdamW` is achieved for $\texttt{lr_ratio} = 16$. Therefore, it will be used in all follow-up experiments.

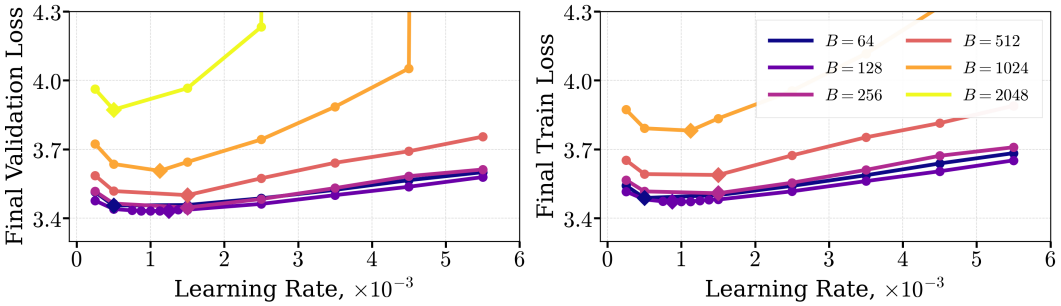


Figure B.2: The final training loss of `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset when varying the batch size and learning rate, while keeping $\texttt{lr_ratio} = 16$, $\alpha = 0.1, \beta = 0.05, \nu = 10^{-8}, \lambda = 0.1$ fixed. Train loss is averaged over 500 iterations.

Fixing $\texttt{lr_ratio} = 16$, we perform additional sweeps over batch size and learning rate with a denser grid for the learning rate around the optimum. We present the results in Figure B.2. We observe that the batch size $B = 128$ is optimal among the tested ones, together the learning rate around 10^{-3}. In follow-up experiments, we set the learning rate $\eta = 0.00125$ based on the validation loss.

Weight Decay Tuning. We next tune weight decay for the base model across token budgets $T \in \{1.3, 10.7, 26.7\} \times 10^9$ tokens. Following our BST scaling rule, we set the batch size according to $B \propto T^{2/3}$, while fixing the learning rate at $\eta = 0.00125$ and the remaining optimizer hyperparameters at $\alpha = 0.1$, $\beta = 0.05$, and $\nu = 10^{-8}$. Thus, weight decay λ is the only optimizer hyperparameter swept in this experiment. As shown in Figure B.3, the training and validation losses are nearly flat around $\lambda = 0.1$, with differences appearing only in the third-fourth decimal place.

We therefore use $\lambda = 0.1$ for the base model, consistent with the strong performance of this choice reported in prior work (Mlodozeniec et al., 2026; Li et al., 2025a; Touvron et al., 2023).

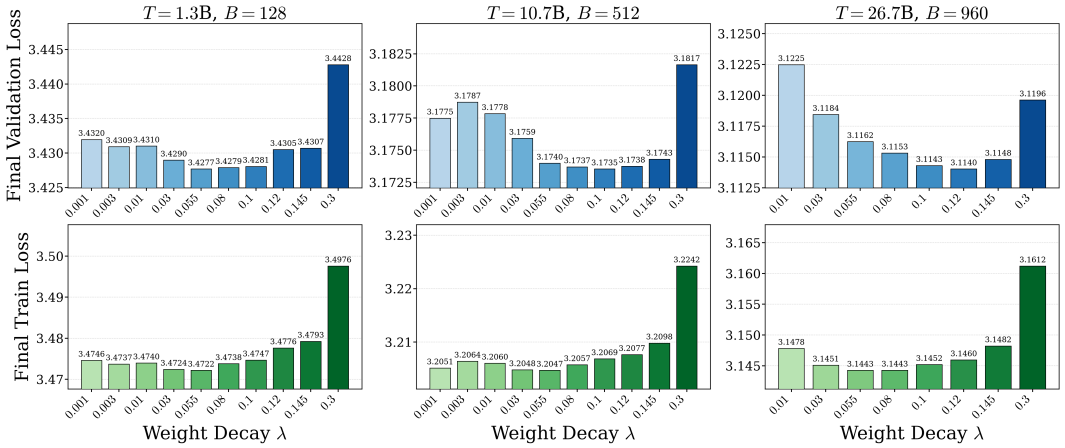


Figure B.3: The final training and validation losses of `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset when varying weight decay, while keeping $B = 128, \eta = 0.00125, \texttt{lr_factor} = 16, \alpha = 0.1, \beta = 0.05, \nu = 10^{-8}$ fixed. Train loss is averaged over 500 iterations.

Momentum Hyperparameter Tuning. Finally, we tune the momentum parameters (α, β) over the Cartesian grid

$$\alpha, \beta \in \{0.2, 0.1, 0.05, 0.025, 0.001\}.$$

Under the conventional notation $\beta_1 = 1 - \alpha$ and $\beta_2 = 1 - \beta$, this corresponds to

$$\beta_1, \beta_2 \in \{0.8, 0.9, 0.95, 0.975, 0.999\}.$$

Throughout the sweep, we fix the remaining hyperparameters to $\nu = 10^{-8}$, $\lambda = 0.1$, $\eta = 0.00125$, $B = 128$, and $\texttt{lr_factor} = 16$. Among all tested combinations, $(\alpha, \beta) = (0.1, 0.05)$ and $(0.05, 0.025)$, corresponding respectively to $(\beta_1, \beta_2) = (0.9, 0.95)$ and $(0.95, 0.975)$, achieve the best performance: the performance of the base model with both pairs is within the noise level. However, since $(\beta_1, \beta_2) = (0.9, 0.95)$ is also widely used in language-model training (Li et al., 2025a; Touvron et al., 2023; Bi et al., 2024; Mehta et al., 2024), we adopt this choice $(\alpha, \beta) = (0.1, 0.05)$ in all subsequent experiments.

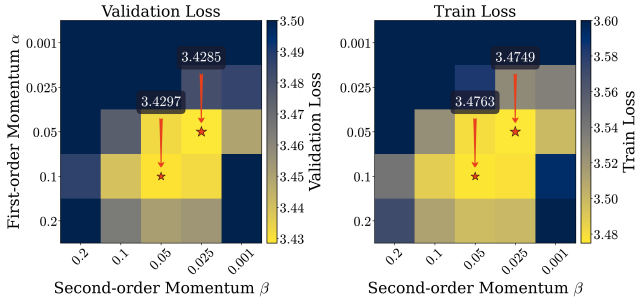


Figure B.4: The final training and validation losses of `AdamW` in the training of the 124M NanoGPT model on the FineWeb dataset when varying momentum hyperparameters α and β, while keeping $\lambda = 0.1, B = 128, \eta = 0.00125, \texttt{lr_factor} = 16, \nu = 10^{-8}$ fixed. Training loss is averaged over 500 iterations.

B.2 Additional Details on the Comparison of Scaling Laws

In the main paper (see Section 6.3), we compare several scaling laws for transferring `AdamW`'s hyperparameters during training of the 124M NanoGPT model from $T = 1.3$B to $T = 40.1$B.

Table B.2: Comparison of scaling laws in the training of the 124M base model. We show the exact choice of hyperparameters at $T_1 = 40.1$B tokens, computed using the scaling laws. If the hyperparameter is not studied in the scaling law, we keep it constant. Otherwise, we follow the prescribed scaling law. Note that in this experiment the model is fixed; therefore, we omit the change with model size.

Reference	Optimizer	Tuned at T_0=1.3B	Scaling Laws	Transfer Rule at T_1=40.1B
Deepseek (Bi et al., 2024)	AdamW	$\eta_0 = 0.00125$, $B = 128$, lr_ratio $= 16$, $\alpha = 0.1, \beta = 0.05$, $\lambda = 0.1$	$B \propto T^{0.3271}, \eta \propto T^{-0.125}$	$\eta_1 = 0.00082, B_1 = 384,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.1$ $^{(a)}$
			$B \propto T^{0.688}, \eta \propto T^{-0.263}$	$\eta_1 = 0.00051, B_1 = 1344,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.1$ $^{(b)}$
Step Law (Li et al., 2025a)			$B \propto T^{0.571}, \eta \propto T^{0.307}$	$\eta_1 = 0.00325, B_1 = 896,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.1$
Power Lines (Bergsma et al., 2025)			$B \propto T^{0.383}, \eta \propto T^{0},$ $\lambda \propto T^{-0.09}$	$\eta_1 = 0.00125, B_1 = 480,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.074$ $^{(c)}$
Complete(d)P (Mlodozeniec et al., 2026)			$m_B \propto T^{1/2}, m_D \propto T$ in Table 1 of Mlodozeniec et al. (2026)	$\eta_1 = 0.000535, B_1 = 704,$ $\alpha = 0.0183, \beta = 0.00917,$ $\lambda = 0.0428$ $^{(d)}$
			$m_B \propto T^{0.44}, m_D \propto T$ in Table 1 of Mlodozeniec et al. (2026)	$\eta_1 = 0.00048, B_1 = 576,$ $\alpha = 0.015, \beta = 0.0075,$ $\lambda = 0.0387$ $^{(e)}$
			$m_B \propto T, m_D \propto T$ in Table 1 of Mlodozeniec et al. (2026)	$\eta_1 = 0.00125, B_1 = 3840,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.1$ $^{(f)}$
BST **[Ours]**			$B \propto T^{2/3}, \lambda \propto T^{-1/3},$ $\alpha, \beta, \eta \propto T^{0}$	$\eta_1 = 0.00125, B_1 = 1248,$ $\alpha = 0.1, \beta = 0.05,$ $\lambda = 0.0321$

$^{(a),(b)}$ The scaling law is originally specified as a function of compute under the joint (model size, training horizon) co-scaling. In $^{(a)}$, we use original $\eta \propto C^{-0.125}$, $B \propto C^{0.3271}$. $^{(b)}$, while in $^{(b)}$, we express the original compute-specific scaling laws under compute-optimal exponents from the same paper.

$^{(c)}$ Weight decay should be adjusted as $\lambda \propto \lambda_0 (T_1/T_0)^{-0.09}$. The authors claim that keeping the learning rate constant while adjusting weight decay is more efficient.

$^{(d),(e),(f)}$ The scaling law for the batch size is not specified and treated as a user choice. In $^{(d)}$, we use the square-root law $B \propto T^{1/2}$ based on derivations of Islamov et al. (2026) (Section C.1). In $^{(e)}$, we use $B \propto T^{0.44}$, the exponent that the follow-up work Bethune et al. (2026) fits for the virtual batch size of a modified rule, under which the learning rate stays constant at this batch size. Finally, in $^{(f)}$, we use the linear scaling $B \propto T$, that is used in the experiments of Figure 14 (Mlodozeniec et al., 2026).

However, some scaling rules might have a different interpretation. Therefore, we provide additional comparisons here. We refer to Figure B.5 for the results. We observe that the DeepSeek scaling law $(\eta \propto T^{-0.125}, B \propto T^{0.3271})$ becomes competitve with our BST rule. However, it is done using a batch size more than three times smaller than that predicted by the BST rule. This might significantly affect the GPU utilization in practice. Other baselines are worse than the BST rule by 0.01 or more in validation loss. In particular, using a linear batch size schedule in Complete(d)P becomes significantly worse than others at long runs (see Figure 14 in Mlodozeniec et al. (2026) for runs around the Chinchilla optimum).

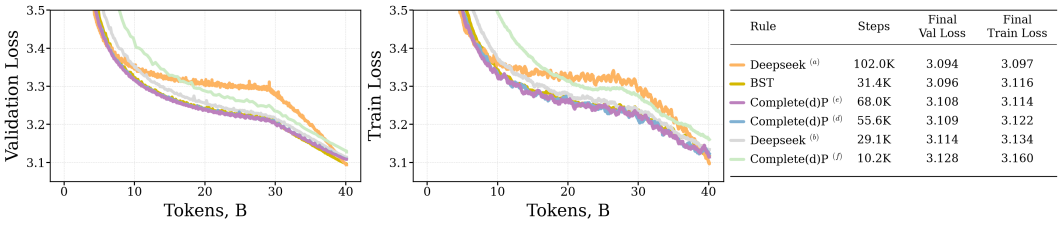


Figure B.5: Comparison of scaling laws for AdamW during training of the 124M NanoGPT model on the FineWeb dataset at a token budget of $T = 40.1$B. We compare additional baselines to the ones presented in Figure 2. Training loss is averaged over 1000 iterations.

C Additional Experiments and Details for Peri-LN model

The experiments with co-scaling of token horizon and model size use the Peri-LN Transformer model (Kim et al., 2025) with several additional changes. In contrast to Pre-LN, Peri-LN places normalization layers after the embedding and not just as the first operation inside the residual blocks, but also as their final operation before adding the block outputs back to the residual stream. Here, we further use μP width and CompleteP depth scaling. Moreover, we use GeLU activation (Hendrycks & Gimpel, 2016) instead of ReLU^2, and untied embedding and head layers. Moreover, the head dimension is 64 rather than 128, so the number of heads grows with d_{model} along the ladder and the per-head geometry is preserved as width increases; Table C.1 gives the resulting shapes. For the optimizer, weight decay is applied in the fully decoupled form of Loshchilov & Hutter (2019), so the quantity tuned and transferred is the product $c = \eta\lambda$ rather than λ itself. Everything else is shared with the model used elsewhere in the paper: rotary position embeddings, RMS normalization of queries and keys before attention, the absence of biases in every linear layer, an MLP ratio of 4, and the GPT-2 vocabulary padded to 50304.

Our optimizer setup includes several modifications motivated by practical considerations. First, we anneal the learning rate to 10% of its peak value, following Hägele et al. (2024), who found that more aggressive annealing can hurt downstream performance. Second, we adopt the $1 - \texttt{sqrt}$ cooldown schedule following the study of Dremov et al. (2025), whose bias–variance analysis identifies this shape as providing a favorable balance between exploration and exploitation. We omit `lr_ratio` because the embedding and output head are untied; determining appropriate groupwise learning-rate scales would require substantially more tuning. We also use μP to avoid the need to use `lr_ratio`. Finally, we use fully decoupled weight decay so that the regularization strength is independent of the learning rate, avoiding confounding between the two hyperparameters (Loshchilov & Hutter, 2019).

Table C.1: The model configurations and training details of Peri-LN (Kim et al., 2025) models with `AdamW`. Width is scaled with μP (base width 256) and depth with CompleteP (base depth 12, $\alpha = 1$).

Hyperparameter	160M	300M	800M	1.4B
Layers	12	16	24	29
Heads	12	16	24	29
Embedding Size	768	1024	1536	1856
Sequence Length	1024			
Weight Tying	No			
Activation Function	GELU			
Vocabulary Size	50304			
Dataset	FineWeb			
Normalization	Peri-LN (RMSNorm), incl. embedding			
Warmup	10% of the total token budget			
Cooldown	20% of the total token budget			
Cooldown Shape	One minus square root			
Final Learning Rate	$0.1\times$ peak			
Stepsize Schedule	WSD			
Gradient Clipping	1.0			
Momentum Parameters	$\alpha = 0.1, \beta = 0.05, (\beta_1 = 0.9, \beta_2 = 0.95)$ *unless the opposite is stated*			
Weight Decay	Fully decoupled with $\eta\lambda = 10^{-4}$ *unless the opposite is stated*			
`lr_ratio`	No manual layer-wise ratio; μP applies $256/d_{\text{model}}$ to matrix parameters and 1.0 to embeddings			
Precision	Mixed precision bf16			
Device Batch Size	largest permissible per scaling law (at most 14)	2	2	16
GPU Type	A100-40GB	A100-40GB	A100-40GB	H200
Code	`https://anonymous.4open.science/r/coscale-gpt-33B4`			

* For the 1.4B model, we have to use a larger device batch size to get the results in a reasonable time.

C.1 Tuning of the Base 162M Model with AdamW

Next, we tune the base configuration from which every scaling law in Section 6.4 transfers. We use the 162M model of Table C.1, with 12 layers, 12 heads, and a hidden dimension of 768. All tuning

Table C.2: Comparison of scaling laws along the co-scaled ladder at $T = 20$B tokens, transferred from the base configuration tuned on the 162M model at $T_0 = 1.62$B tokens (Section C.1), so $T/T_0 = 12.3$ for every model size. We show each scaling law in model size d and token budget T, and the resulting hyperparameters. A hyperparameter that a scaling law does not study is kept at its base value.

Reference	Base at 162M	Scaling Laws	162M	304M	834M	1.39B
BST **[Ours]**	$\eta_0 = 2.19 \times 10^{-2}$, $B_0 = 128$, $c_0 = \eta_0\lambda_0 = 10^{-4}$, $\alpha = 0.1$, $\beta = 0.05$, $\nu = 10^{-14}$	$B \propto T^{2/3}$, $\eta \propto d^0T^0$, $c \propto d^{-1/2}T^{-1/3}$ [a]	$\eta =$ 2.19e-2 $B = 688$ $c =$ 4.33e-5	2.19e-2 688 3.16e-5	2.19e-2 688 1.91e-5	2.19e-2 688 1.48e-5
Power Lines (Bergsma et al., 2025)		$B \propto T^{0.383}$, $\eta \propto d^0T^0$, $c \propto d^{-0.527}T^{-0.09}$	$\eta =$ 2.19e-2 $B = 336$ $c =$ 7.98e-5	2.19e-2 336 5.73e-5	2.19e-2 336 3.37e-5	2.19e-2 336 2.58e-5
Step Law (Li et al., 2025a)		$B \propto T^{0.571}$, $\eta \propto d_{\text{ne}}^{-0.713}T^{0.307}$	$\eta =$ 4.74e-2 $B = 544$	2.56e-2 544	1.08e-2 544	7.18e-3 544
Complete(d)P (Mlodozeniec et al., 2026)		$B \propto T^{0.44}$, $\eta \propto (B/T)^{1/2}$, [b] $c \propto \eta$, $\alpha, \beta \propto B/T$	$\eta =$ 1.08e-2 $B = 384$ $c =$ 4.93e-5,	1.08e-2 384 $(\beta_1, \beta_2) = (0.9757, 0.9878)$	1.08e-2 384	1.08e-2 384
DeepSeek (Bi et al., 2024)		$B \propto C^{0.3271}$, $\eta \propto C^{-0.125}$, [c] $C = MT$	$\eta =$ 1.60e-2 $B = 288$	1.45e-2 384	1.25e-2 560	1.17e-2 672

[a] d is the total parameter count; d_{ne} excludes the embedding and the output layer.
[b] The batch-size law is left to the user; we use $B \propto T^{0.44}$ of Bethune et al. (2026), as in [e] of Table B.2.
[c] DeepSeek's scaling law is fitted against compute $C = MT$ with M the non-embedding FLOPs per token, so its model-size dependence is via M rather than N. It is the only rule whose batch size varies along the ladder, because it is the only one whose batch law has a model-size term.

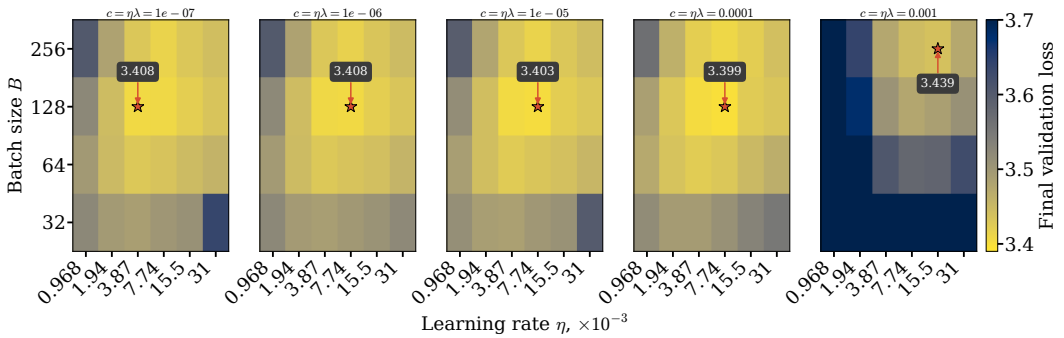


Figure C.1: The final validation loss of `AdamW` in the training of the 162M model on the FineWeb dataset at $T = 1.62$B tokens, when varying the batch size, the learning rate and the decoupled weight decay $c = \eta\lambda$, while keeping $\alpha = 0.1$, $\beta = 0.05$ and $\nu = 10^{-14}$ fixed and Peri-LN disabled. The grid selects $\eta_{\text{base}} = 7.74 \times 10^{-3}$, which is the learning rate the Peri-LN sweep of Figure C.2 starts from.

runs use a budget of $T_0 = 1.62 \times 10^9$ tokens with sequence length 1024, approximately 10 tokens per parameter. Since weight decay is fully decoupled, the tuned quantity is $c = \eta\lambda$.

Batch Size and Learning Rate Tuning. Since μP fixes the relative learning rate of the embedding, head layers, and other layers, there is no `lr_ratio` to tune, and we sweep the three of the remaining hyperparameters jointly over

$$B \in \{32, 64, 128, 256\},$$
$$\eta \in \{0.125, 0.25, 0.5, 1, 2, 4\} \times \eta_{\text{base}},$$
$$c = \eta\lambda \in \{10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\},$$

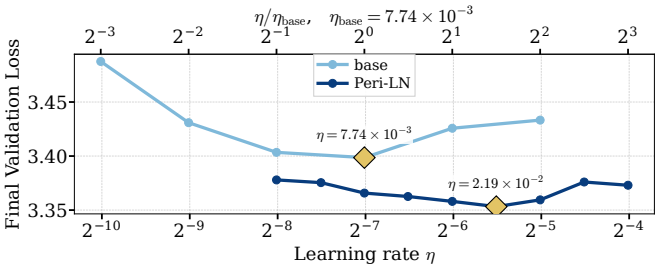


Figure C.2: The learning-rate sweep at $B = 128$ and $c = 10^{-4}$, with and without Peri-LN, at $T = 1.62$B tokens. Peri-LN moves the optimum from $\eta_{\text{base}} = 7.74 \times 10^{-3}$ to $\eta_0 = 2.19 \times 10^{-2}$.

keeping $\alpha = 0.1$, $\beta = 0.05$ and $\nu = 10^{-14}$ fixed. We report this grid in Figure C.1. The best configuration is $B = 128$, $\eta_{\text{base}} = 7.74 \times 10^{-3}$ and $c = 10^{-4}$, at a validation loss of 3.3986, and the minimum is interior in all three axes.

Peri-LN Learning Rate Tuning. The grid above is run without Peri-LN, so the learning rate it selects need not remain optimal once Peri-LN is added; we therefore re-tune it. As shown in Figure C.2, the optimum moves by a factor of 2.83, from $\eta_{\text{base}} = 7.74 \times 10^{-3}$ to $\eta_0 = 2.19 \times 10^{-2}$. Since every model in Section 6.4 uses Peri-LN, the base learning rate used there is η_0.

Other Hyperparameters. We do not repeat the momentum grid search at this model size. Section B.1 selects $(\alpha, \beta) = (0.1, 0.05)$, that is $(\beta_1, \beta_2) = (0.9, 0.95)$, and these values are widely used in language-model pre-training. We keep them and set $\nu = 10^{-14}$ with gradient clipping at 1.0.

Schedule. All runs use a WSD schedule with 10% of the budget in warmup and 20% in a $1-\texttt{sqrt}$ cooldown to a final learning rate of 0.1η, following Hägele et al. (2024).

The base set of hyparameters used in Section 6.4 is thus

$$B_0 = 128, \quad \eta_0 = 2.19 \times 10^{-2}, \quad c_0 = \eta_0 \lambda_0 = 10^{-4}, \quad (\beta_1, \beta_2) = (0.9, 0.95), \quad \nu = 10^{-14},$$

at $T_0 = 1.62 \times 10^9$ tokens.

C.2 Coordinate Checks and Problem Constants for the Co-scaled Ladder

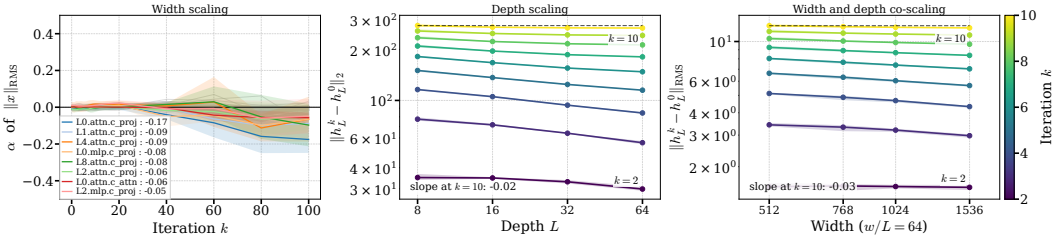


Figure C.3: Coordinate checks for the 162M Peri-LN model at the tuned learning rate. **Left:** the width exponents α of the activation RMS norm for every layer output against the iteration, with bands over four seeds and the eight largest plotted. **Middle and right:** the residual stream update at iterations $k = 2, \ldots, 10$ against depth, following Figure 7 of Dey et al. (2025), and against width with depth co-scaled at a width-to-depth ratio of 64. A stable parameterization keeps these flat (dashed). The right panel uses the RMS norm rather than the Frobenius norm of Dey et al. (2025), since d_{model} varies along that axis.

Coordinate Checks. The ladder of Section 6.4 transfers one learning rate across a $8.6\times$ range of model sizes, which assumes that the parameterization keeps the amount of feature learning invariant from small to large width and depth. We verify this in Figure C.3, in three co-scaling tests. For pure width scaling, μP should preserve the activation RMS norm across model scales, that is keep α close

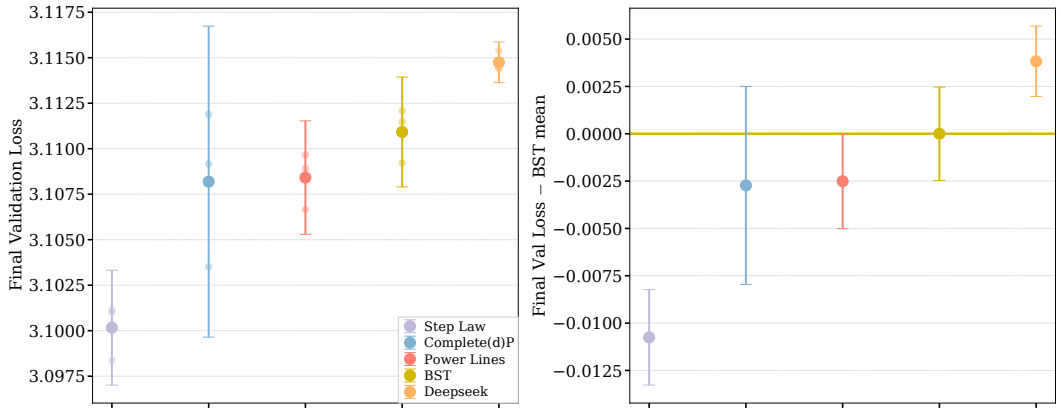
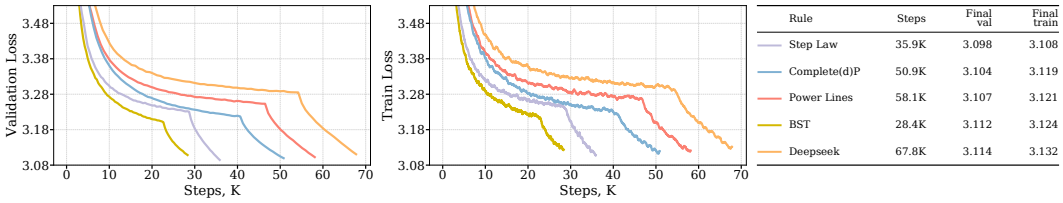


Figure C.4: Seed sensitivity at the 162M model, with three seeds per scaling law varying the initialization and the data order together. **Left:** the individual runs with their mean and ± 2 standard deviations. **Right:** the difference from the BST mean, with ± 2 standard deviations of that difference. The standard deviation of the paired difference $\mathrm{rule}_i - \mathrm{BST}_i$ is 3.27×10^{-3}, giving the $\pm 6.5 \times 10^{-3}$ band in Figure 1.

to 0. For depth and for width and depth co-scaled at the ladder's aspect ratio, CompleteP's Maximal Residual Stream Update desideratum requires the residual stream update to remain flat across model scales. The slopes over $k = 2, \ldots, 10$ are indeed near-zero with 0.086 and 0.087.

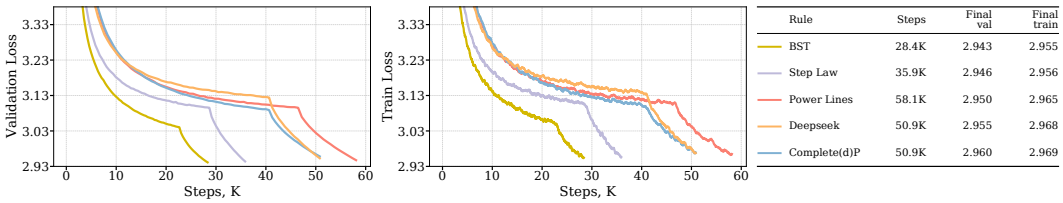
Seed Sensitivity. The differences in Figure 1 are of the order of 10^{-3} to 10^{-2}, so we measure the seed sensitivity. In Figure C.4, we run every scaling law at the 162M model with three seeds, jointly varying the initialization seed and the data-order seed. We then compare in Figure 1 the difference between two runs, and we measure its standard deviation as 3.27×10^{-3}. The band drawn in Figure 1 is two standard deviations, corresponding to $\pm 6.5 \times 10^{-3}$.

C.3 Loss Dynamics Along the Ladder



Rule	Steps	Final val	Final train
Step Law	35.9K	3.098	3.108
Complete(d)P	50.9K	3.104	3.119
Power Lines	58.1K	3.107	3.121
BST	28.4K	3.112	3.124
Deepseek	67.8K	3.114	3.132

Figure C.5: Training dynamics of the scaling rules at the 162M model, $T = 20$B tokens. From the left: validation loss and training loss against tokens, then the final validation and training losses. Training loss is averaged over 1% of each run's steps, and the final validation loss is the run's full-split evaluation.



Rule	Steps	Final val	Final train
BST	28.4K	2.943	2.955
Step Law	35.9K	2.946	2.956
Power Lines	58.1K	2.950	2.965
Deepseek	50.9K	2.955	2.968
Complete(d)P	50.9K	2.960	2.969

Figure C.6: As Figure C.5, at the 304M model.

Figure 1 compares only the final losses. Here, in Figures C.5 to C.8, we provide the full trajectories at each model size. The horizontal axis is in tokens. The ranking of scaling laws is established early

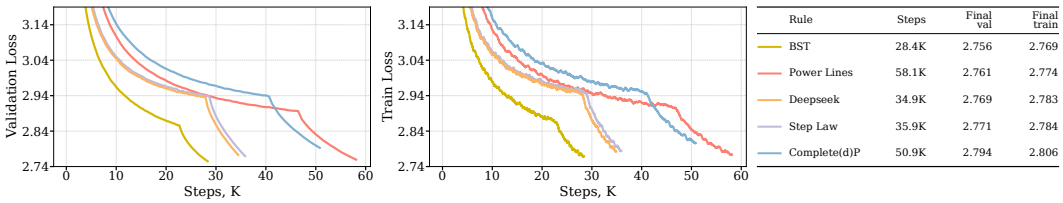


Figure C.7: As Figure C.5, at the 834M model.

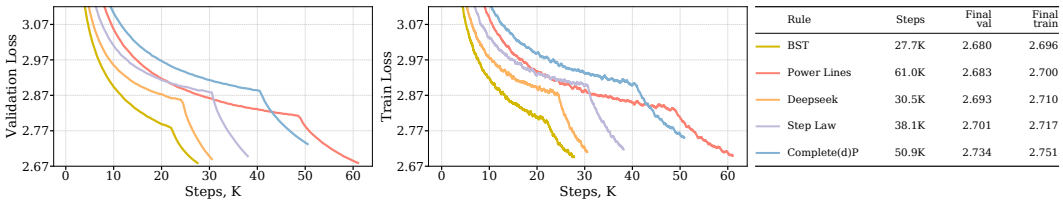


Figure C.8: As Figure C.5, at the 1.4B model.

and change little after the first few billion tokens, except at the cooldown, where the curves converge; Complete(d)P is the exception, separating from the others progressively as the model grows.

C.4 MFU AND THROUGHPUT

The scaling rules in Table 1 prescribe different global batch sizes, and a larger batch needs fewer optimizer steps for the same token budget (Section C.3). Whether this also shortens training depends on how the batch is split across devices, which Figure C.9 measures for the five model sizes of the ladder. The model FLOPs utilization (MFU) rises by a factor of 2 to 5 as the per-GPU micro-batch grows to the memory limit, and gradient accumulation at that micro-batch adds a further 2-18% before saturating at a global batch of about 128-256 sequences, below every rule's batch size. At a fixed number of GPUs, a larger batch therefore does not process tokens faster; its advantage is that it can be spread over more GPUs at a saturated per-GPU batch, which reduces the number of sequential optimizer steps and hence the wall-clock time. All numbers are for eager PyTorch, as used in training; `torch.compile` raises MFU by 1.7-1.9× for the three smaller models and by about 1.4× for the two larger ones.

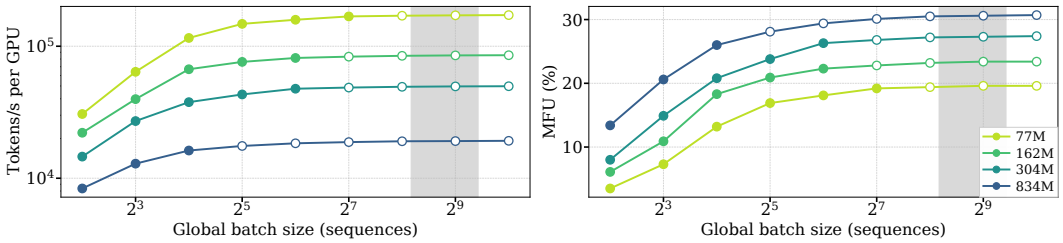


Figure C.9: **Hardware efficiency is set by the per-GPU micro-batch, not by the global batch size.** Throughput (left) and model FLOPs utilization (MFU, right) of `AdamW` training on A100-40GB GPUs against the global batch size, for four model sizes of the ladder (bf16, sequence length 1024, data parallel over 4 GPUs). Filled markers grow the per-GPU micro-batch without gradient accumulation; hollow markers accumulate gradients at the largest micro-batch that fits in memory. MFU counts $6N_{\text{matmul}} + 12LdS$ FLOPs per token against the 312 TFLOP/s bf16 peak. The grey band marks the global batch sizes the scaling rules prescribe on the ladder.

D CONVERGENCE ANALYSIS OF SGD WITH MOMENTUM

Table D.1: Notation used in the analysis of SGD with momentum algorithm.

Notation	Meaning	Interpretation
e_k	$\mathbb{E}\left[\left\|m_k - \nabla f(x_k)\right\|^2\right]$	Momentum error
F_k	$\mathbb{E}[f(x_k) - f^\star]$	Expected function sub-optimality
V_k	$\frac{5\eta}{\alpha} e_k$	Expected momentum error
κ_ε	$\min\left\{\frac{\alpha}{5}, \frac{\eta\mu^2\varepsilon}{8}\right\}$	Lyapunov function contraction factor
Φ_k	$f(x_k) - f^\star + \frac{5\eta}{\alpha}\|m_k - \nabla f(x_k)\|_2^2$	Lyapunov function
Y_k	$(\mathbb{E}[\Phi_k] - \varepsilon/2)_+$	Modified Lyapunov function

Algorithm 1 SGD with Momentum

1: **Input:** Initial point x_0, hyperparameters $\eta > 0$, $\alpha \in (0, 1]$, momentum buffer $m_0 = 0$
2: **for** $k = 0, 2, \ldots, K-1$ **do**
3: Compute stochastic gradient g_k at x_k
4: Update momentum $m_k = (1-\alpha)m_{k-1} + \alpha g_k$
5: Update parameters $x_{k+1} = x_k - \eta m_k$
6: **end for**

Our analysis assumes that Assumptions 1 and 2 hold on a compact set containing the entire SGDM trajectory.

D.1 AUXILIARY LEMMAS

Lemma 1. *Under Assumption 1, for $\eta \le \frac{1}{L}$, the SGDM iterates satisfy*

$$f(x_{k+1}) \le f(x_k) - \frac{\eta}{2}\left\|\nabla f(x_k)\right\|^2 + \frac{\eta}{2}\left\|m_k - \nabla f(x_k)\right\|^2.$$

Proof. By L-smoothness (Assumption 1),

$$f(x_{k+1}) \le f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k\rangle + \frac{L}{2}\left\|x_{k+1} - x_k\right\|^2.$$

Substituting $x_{k+1} - x_k = -\eta m_k$,

$$f(x_{k+1}) \le f(x_k) - \eta\langle \nabla f(x_k), m_k\rangle + \frac{L\eta^2}{2}\left\|m_k\right\|^2.$$

Applying the polarisation identity $\langle a, b\rangle = \frac{1}{2}(\|a\|^2 + \|b\|^2 - \|a-b\|^2)$,

$$\begin{aligned} f(x_{k+1}) &\le f(x_k) - \frac{\eta}{2}\left\|\nabla f(x_k)\right\|^2 - \frac{\eta}{2}\left\|m_k\right\|^2 + \frac{\eta}{2}\left\|m_k - \nabla f(x_k)\right\|^2 + \frac{L\eta^2}{2}\left\|m_k\right\|^2 \\ &= f(x_k) - \frac{\eta}{2}\left\|\nabla f(x_k)\right\|^2 + \frac{\eta}{2}\left\|m_k - \nabla f(x_k)\right\|^2 + \frac{\eta}{2}(L\eta - 1)\left\|m_k\right\|^2. \end{aligned}$$

Since $\eta \le \frac{1}{L}$ implies $L\eta - 1 \le 0$, the last term is non-positive, and we obtain

$$f(x_{k+1}) \le f(x_k) - \frac{\eta}{2}\left\|\nabla f(x_k)\right\|^2 + \frac{\eta}{2}\left\|m_k - \nabla f(x_k)\right\|^2. \qquad \square$$

Lemma 2. *Under Assumption 1 and 3, let*

$$e_k \coloneqq \mathbb{E}\left[\left\|m_k - \nabla f(x_k)\right\|^2\right].$$

Then, for $1 \ge \alpha \ge 8L\eta$,

$$e_k \le \left(1 - \frac{2\alpha}{5}\right) e_{k-1} + \frac{6L^2\eta^2}{\alpha}\mathbb{E}\left[\left\|\nabla f(x_{k-1})\right\|^2\right] + 2\alpha^2\sigma^2.$$

Proof. From the momentum update $m_k = (1-\alpha)m_{k-1} + \alpha g_k$:

$$m_k - \nabla f(x_k) = (1-\alpha)\big(m_{k-1} - \nabla f(x_k)\big) + \alpha\big(g_k - \nabla f(x_k)\big).$$

Squaring and expanding:

$$\begin{aligned}\|m_k - \nabla f(x_k)\|^2 = (1-\alpha)^2 \|m_{k-1} - \nabla f(x_k)\|^2 \\ + 2\alpha(1-\alpha)\big\langle m_{k-1} - \nabla f(x_k), g_k - \nabla f(x_k)\big\rangle \\ + \alpha^2 \|g_k - \nabla f(x_k)\|^2 .\end{aligned}$$

By Assumption 3 and the tower property, the cross term satisfies:

$$\begin{aligned}\mathbb{E}\left[\big\langle m_{k-1} - \nabla f(x_k), g_k - \nabla f(x_k)\big\rangle\right] &= \mathbb{E}\left[\big\langle m_{k-1} - \nabla f(x_k), \mathbb{E}\big[g_k - \nabla f(x_k) \mid \mathcal{F}_k\big]\big\rangle\right] \\ &= 0.\end{aligned}$$

Taking full expectation and applying Assumption 3:

$$\begin{aligned}e_k &= (1-\alpha)^2 \mathbb{E}\left[\|m_{k-1} - \nabla f(x_k)\|^2\right] + \alpha^2 \mathbb{E}\left[\|g_k - \nabla f(x_k)\|^2\right] \\ &\le (1-\alpha)^2 \mathbb{E}\left[\|m_{k-1} - \nabla f(x_k)\|^2\right] + \alpha^2\sigma^2.\end{aligned}$$

Since $m_{k-1} - \nabla f(x_k) = \big(m_{k-1} - \nabla f(x_{k-1})\big) + \big(\nabla f(x_{k-1}) - \nabla f(x_k)\big)$, applying Young's inequality $\|a+b\|^2 \le (1+t)\|a\|^2 + (1+\frac{1}{t})\|b\|^2$ with $t = \frac{\alpha}{2}$:

$$\begin{aligned}\|m_{k-1} - \nabla f(x_k)\|^2 \le \left(1 + \frac{\alpha}{2}\right) \|m_{k-1} - \nabla f(x_{k-1})\|^2 \\ + \left(1 + \frac{2}{\alpha}\right) \|\nabla f(x_{k-1}) - \nabla f(x_k)\|^2 .\end{aligned}$$

By L-smoothness and $x_k - x_{k-1} = -\eta m_{k-1}$, then $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$:

$$\begin{aligned}\|\nabla f(x_{k-1}) - \nabla f(x_k)\|^2 &\le L^2\eta^2 \|m_{k-1}\|^2 \\ &= L^2\eta^2 \left\|\big(m_{k-1} - \nabla f(x_{k-1})\big) + \nabla f(x_{k-1})\right\|^2 \\ &\le 2L^2\eta^2 \|m_{k-1} - \nabla f(x_{k-1})\|^2 + 2L^2\eta^2 \|\nabla f(x_{k-1})\|^2 .\end{aligned}$$

Substituting into the Young's bound and then into the e_k inequality:

$$\begin{aligned}e_k \le (1-\alpha)^2 \left[\left(1 + \frac{\alpha}{2}\right) + 2L^2\eta^2 \left(1 + \frac{2}{\alpha}\right)\right] e_{k-1} \\ + 2(1-\alpha)^2 \left(1 + \frac{2}{\alpha}\right) L^2\eta^2 \mathbb{E}\left[\|\nabla f(x_{k-1})\|^2\right] \\ + \alpha^2\sigma^2.\end{aligned}$$

Since $(1-\alpha)^2 \le 1$ and $\frac{\alpha+2}{\alpha} \le \frac{3}{\alpha}$ for $\alpha \le 1$:

$$2(1-\alpha)^2 \left(1 + \frac{2}{\alpha}\right) L^2\eta^2 \le \frac{6L^2\eta^2}{\alpha}.$$

Since $\alpha \ge 8L\eta$ gives $L^2\eta^2 \le \frac{\alpha^2}{64}$:

$$\begin{aligned}2L^2\eta^2 \left(1 + \frac{2}{\alpha}\right) \le \frac{\alpha^2}{32} \cdot \frac{\alpha+2}{\alpha} \le \frac{\alpha}{16} + \frac{\alpha^2}{32}, \\ \left(1 + \frac{\alpha}{2}\right) + 2L^2\eta^2 \left(1 + \frac{2}{\alpha}\right) \le 1 + \frac{9\alpha}{16} + \frac{\alpha^2}{32}.\end{aligned}$$

Since $(1-\alpha)^2 \le 1 - \alpha$ for $\alpha \in [0,1]$:

$$\begin{aligned}(1-\alpha)^2 \left(1 + \frac{9\alpha}{16} + \frac{\alpha^2}{32}\right) \le (1-\alpha) \left(1 + \frac{9\alpha}{16}\right) \\ = 1 - \frac{7\alpha}{16} - \frac{9\alpha^2}{16} \le 1 - \frac{7\alpha}{16} \le 1 - \frac{2\alpha}{5}.\end{aligned}$$

Combining and using $\alpha^2\sigma^2 \le 2\alpha^2\sigma^2$:

$$e_k \le \left(1 - \frac{2\alpha}{5}\right) e_{k-1} + \frac{6L^2\eta^2}{\alpha} \mathbb{E}\left[\|\nabla f(x_{k-1})\|^2\right] + 2\alpha^2\sigma^2. \qquad \square$$

Lemma 3. *Under Assumptions Assumption 1, 2, and 3, let*

$$\Phi_k \coloneqq f(x_k) - f^\star + \frac{5\eta}{\alpha} e_k, \qquad V_k \coloneqq \frac{5\eta}{\alpha}\mathbb{E}\left[e_k\right], \qquad F_k \coloneqq \mathbb{E}[f(x_k) - f^\star].$$

For $\eta \le \frac{1}{L}$ and $1 \ge \alpha \ge 2\sqrt{30}L\eta$,

$$\mathbb{E}[\Phi_{k+1}] \le \mathbb{E}[\Phi_k] - \frac{\alpha}{5} V_k - \frac{\eta\mu^2}{4} F_k^2 + 10\eta\alpha\sigma^2. \tag{1}$$

Proof. Taking expectation of Lemma 1 and using $\mathbb{E}[e_k] = e_k$:

$$\mathbb{E}[f(x_{k+1}) - f^\star] \le \mathbb{E}[f(x_k) - f^\star] - \frac{\eta}{2}\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + \frac{\eta}{2} e_k.$$

Multiplying Lemma 2 by $\frac{5\eta}{\alpha}$:

$$\frac{5\eta}{\alpha} e_{k+1} \le \left(1 - \frac{2\alpha}{5}\right)\frac{5\eta}{\alpha} e_k + \frac{30L^2\eta^3}{\alpha^2}\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + 10\eta\alpha\sigma^2.$$

Adding these two inequalities and using the definition of Φ_k:

$$\begin{aligned}
\mathbb{E}[\Phi_{k+1}] &\le \mathbb{E}[f(x_k) - f^\star] - \frac{\eta}{2}\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + \frac{\eta}{2} e_k \\
&\quad + \left(1 - \frac{2\alpha}{5}\right)\frac{5\eta}{\alpha} e_k + \frac{30L^2\eta^3}{\alpha^2}\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + 10\eta\alpha\sigma^2 \\
&= \mathbb{E}[f(x_k) - f^\star] + \left(1 - \frac{\alpha}{5}\right)\frac{5\eta}{\alpha} e_k \\
&\quad - \frac{\eta}{2}\left(1 - \frac{60L^2\eta^2}{\alpha^2}\right)\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + 10\eta\alpha\sigma^2.
\end{aligned}$$

Since $\alpha \ge 2\sqrt{30}L\eta$ implies $\frac{60L^2\eta^2}{\alpha^2} \le \frac{1}{2}$:

$$\mathbb{E}[\Phi_{k+1}] \le \mathbb{E}[f(x_k) - f^\star] + \left(1 - \frac{\alpha}{5}\right)\frac{5\eta}{\alpha} e_k - \frac{\eta}{4}\mathbb{E}\left[\left\|\nabla f(x_k)\right\|^2\right] + 10\eta\alpha\sigma^2.$$

By Assumption 2, $\left\|\nabla f(x_k)\right\|^2 \ge \mu^2 (f(x_k) - f^\star)^2$:

$$\mathbb{E}[\Phi_{k+1}] \le \mathbb{E}[f(x_k) - f^\star] + \left(1 - \frac{\alpha}{5}\right)\frac{5\eta}{\alpha} e_k - \frac{\eta\mu^2}{4}\mathbb{E}\left[(f(x_k) - f^\star)^2\right] + 10\eta\alpha\sigma^2.$$

By Jensen's inequality $\mathbb{E}[X^2] \ge (\mathbb{E}[X])^2$:

$$\begin{aligned}
\mathbb{E}[\Phi_{k+1}] &\le \mathbb{E}[f(x_k) - f^\star] + \left(1 - \frac{\alpha}{5}\right)\frac{5\eta}{\alpha} e_k - \frac{\eta\mu^2}{4} F_k^2 + 10\eta\alpha\sigma^2 \\
&= \mathbb{E}[\Phi_k] - \frac{\alpha}{5} V_k - \frac{\eta\mu^2}{4} F_k^2 + 10\eta\alpha\sigma^2.
\end{aligned}$$

□

Lemma 4. *Under the notation of Lemma 3, let $\varepsilon > 0$ and define*

$$\kappa_\varepsilon \coloneqq \min\left\{\frac{\alpha}{5}, \frac{\eta\mu^2\varepsilon}{8}\right\}, \qquad Y_k \coloneqq \left(\mathbb{E}[\Phi_k] - \varepsilon/2\right)_+.$$

Then, we have

$$Y_K \le \left(1 - \kappa_\varepsilon\right)^K Y_0 + \frac{10\eta\alpha\sigma^2}{\kappa_\varepsilon}.$$

Proof. We first prove the auxiliary inequality

$$\frac{\alpha}{5} V_k + \frac{\eta\mu^2}{4} F_k^2 \ge \kappa_\varepsilon (\mathbb{E}[\Phi_k] - \varepsilon/2)_+. \tag{7}$$

Case 1: $F_k \geq \varepsilon/2$**.**

$$F_k^2 \geq \frac{\varepsilon}{2} F_k \Longrightarrow \frac{\eta\mu^2}{4} F_k^2 \geq \frac{\eta\mu^2\varepsilon}{8} F_k \geq \kappa_\varepsilon F_k,$$
$$\frac{\alpha}{5} V_k \geq \kappa_\varepsilon V_k,$$
$$\frac{\alpha}{5} V_k + \frac{\eta\mu^2}{4} F_k^2 \geq \kappa_\varepsilon (V_k + F_k) = \kappa_\varepsilon \mathbb{E}[\Phi_k] \geq \kappa_\varepsilon (\mathbb{E}[\Phi_k] - \varepsilon/2)_+ = \kappa_\varepsilon Y_k.$$

Case 2: $F_k \leq \varepsilon/2$**.**

$$(\mathbb{E}[\Phi_k] - \varepsilon/2)_+ = (F_k + V_k - \varepsilon/2)_+ \leq V_k,$$
$$\frac{\alpha}{5} V_k + \frac{\eta\mu^2}{4} F_k^2 \geq \frac{\alpha}{5} V_k \geq \kappa_\varepsilon V_k \geq \kappa_\varepsilon (\mathbb{E}[\Phi_k] - \varepsilon/2)_+.$$

In both cases, (7) holds. Combining with Lemma 3:

$$\begin{aligned}\mathbb{E}[\Phi_{k+1}] &\leq \mathbb{E}[\Phi_k] - \frac{\alpha}{5} V_k - \frac{\eta\mu^2}{4} F_k^2 + 10\eta\alpha\sigma^2 \\ &\leq \mathbb{E}[\Phi_k] - \kappa_\varepsilon (\mathbb{E}[\Phi_k] - \varepsilon/2)_+ + 10\eta\alpha\sigma^2.\end{aligned}$$

Now we subtract $\varepsilon/2$ from both sides and take the positive parts on both sides. Using inequalities $(s + c)_+ \leq (s)_+ + c$, $(s - \kappa_\varepsilon s)_+ \leq (1 - \kappa_\varepsilon)s$ for $c > 0, s \geq 0$ and $\kappa_\varepsilon \leq 1$:

$$\begin{aligned}Y_{k+1} &= (\mathbb{E}[\Phi_{k+1}] - \varepsilon/2)_+ \\ &\leq \left(\mathbb{E}[\Phi_k] - \varepsilon/2 - \kappa_\varepsilon (\mathbb{E}[\Phi_k] - \varepsilon/2)_+ + 10\eta\alpha\sigma^2\right)_+ \\ &\leq (1 - \kappa_\varepsilon) Y_k + 10\eta\alpha\sigma^2.\end{aligned}$$

Applying this one-step contraction repeatedly and summing the geometric series:

$$\begin{aligned}Y_K &\leq (1 - \kappa_\varepsilon)^K Y_0 + 10\eta\alpha\sigma^2 \sum_{j=0}^{K-1} (1 - \kappa_\varepsilon)^j \\ &\leq (1 - \kappa_\varepsilon)^K Y_0 + \frac{10\eta\alpha\sigma^2}{\kappa_\varepsilon}.\end{aligned}$$

□

D.2 Proof of the Main Theorem

Theorem 4 (Full statement of Theorem 1)**.** *Under Assumptions 1, 2, 3, let $\varepsilon > 0$ and set*

$$\alpha = \min\left\{1, \frac{\varepsilon^2\mu^2}{320\sigma^2}\right\}, \quad \eta = \min\left\{\frac{1}{\sqrt{120}L}, \frac{(\varepsilon\mu)^2}{320\sqrt{120}L\sigma^2}, \frac{8}{5\mu^2\varepsilon}, \frac{\varepsilon}{200\sigma^2}\right\},$$

and

$$K = \max\left\{5, \frac{8\sqrt{120}L}{\mu^2\varepsilon}, \frac{320\sqrt{120}L\sigma^2}{\mu^4\varepsilon^3}, \frac{1600\sigma^2}{\varepsilon^2\mu^2}\right\} \log\left(\frac{4Y_0}{\varepsilon}\right).$$

Then, we have $\mathbb{E}[f(x_K) - f^\star] \leq \varepsilon$

Proof. The choice of η ensures that $\frac{\alpha}{5} \geq \frac{\eta\mu^2\varepsilon}{8}$, i.e., $\kappa_\varepsilon = \frac{\eta\mu^2\varepsilon}{8}$. Then, the bound from Lemma 4 becomes

$$Y_K \leq \left(1 - \frac{\eta\mu^2\varepsilon}{8}\right)^K Y_0 + \frac{10\eta\alpha\sigma^2}{\eta\mu^2\varepsilon/8} = \left(1 - \frac{\eta\mu^2\varepsilon}{8}\right)^K Y_0 + \frac{80\alpha\sigma^2}{\mu^2\varepsilon}. \tag{8}$$

Next, the choice of $\alpha \leq \frac{\varepsilon^2\mu^2}{320\sigma^2}$ ensures that

$$\alpha \leq \frac{\varepsilon^2\mu^2}{320\sigma^2} \Longleftrightarrow \frac{80\alpha\sigma^2}{\mu^2\varepsilon} \leq \frac{\varepsilon}{4}.$$

To satisfy the constraints $\alpha = \sqrt{120}L\eta$ and $\eta \leq \frac{1}{L}$ from Lemma 3, we choose $\eta \leq \min\left\{\frac{1}{L}, \frac{\alpha}{\sqrt{120}L}\right\}$.

We now want to make the first term in (8) to be at most $\varepsilon/4$:

$$\left(1 - \frac{\eta\mu^2\varepsilon}{8}\right)^K Y_0 \leq \frac{\varepsilon}{4},$$

which is satisfied for K such that

$$K \geq \frac{8}{\eta\mu^2\varepsilon}\log\left(\frac{4Y_0}{\varepsilon}\right).$$

Combining two bounds together, we have $Y_K \leq \varepsilon/4 + \varepsilon/4 = \varepsilon/2$, hence

$$Y_K \leq \varepsilon/2 \Rightarrow \mathbb{E}[\Phi_K] \leq \varepsilon/2 + \varepsilon/2 = \varepsilon \Rightarrow \mathbb{E}[f(x_K) - f^\star] \leq \varepsilon.$$

When $\alpha = 1$ (large-ε regime),

$$\eta = \min\left\{\frac{1}{\sqrt{120}L}, \frac{8}{5\mu^2\varepsilon}\right\} \quad \text{and} \quad K = \max\left\{\frac{8\sqrt{120}L}{\mu^2\varepsilon}, 5\right\}\log\left(\frac{4Y_0}{\varepsilon}\right).$$

When $\alpha = \frac{\varepsilon^2\mu^2}{320\sigma^2}$ (small-ε regime),

$$\eta = \left\{\frac{\varepsilon^2\mu^2}{320\sigma^2\sqrt{120}L}, \frac{\varepsilon}{200\sigma^2}\right\} \quad \text{and} \quad K = \max\left\{\frac{320\sqrt{120}L\sigma^2}{\mu^4\varepsilon^3}, \frac{1600\sigma^2}{\varepsilon^2\mu^2}\right\}\log\left(\frac{4Y_0}{\varepsilon}\right).$$

Grouping two cases together gives

$$\alpha = \min\left\{1, \frac{(\varepsilon\mu)^2}{320\sigma^2}\right\}, \quad \eta = \min\left\{\frac{1}{\sqrt{120}L}, \frac{(\varepsilon\mu)^2}{320\sqrt{120}L\sigma^2}, \frac{8}{5\mu^2\varepsilon}, \frac{\varepsilon}{200\sigma^2}\right\},$$

and

$$K = \max\left\{5, \frac{8\sqrt{120}L}{\mu^2\varepsilon}, \frac{320\sqrt{120}L\sigma^2}{\mu^4\varepsilon^3}, \frac{1600\sigma^2}{\varepsilon^2\mu^2}\right\}\log\left(\frac{4Y_0}{\varepsilon}\right)$$

□

D.3 Error Floor and BST Rule

Corollary 3 (Full statement of Corollary 1)**.** *Let the hyperparameters of Algorithm 1 be chosen according to Theorem 4. Then after T/B iterations, we achieve the optimization error*

$$\varepsilon = \tilde{\mathcal{O}}\left(\max\left\{\frac{LB}{\mu^2 T}, \frac{\sigma_\star}{\mu\sqrt{T}}, \left(\frac{\sigma_\star^2 L}{\mu^4 T}\right)^{1/3}\right\}\right),$$

where $\widetilde{\mathcal{O}}$ hides numerical constant and logarithmic factors. If $T \geq \mathcal{O}\left(\left(\frac{\mu\sigma_\star}{L}\right)^2\right)$, then the BST scaling rule for SGDM *is given by*

$$B \propto T^{2/3} \cdot \left(\frac{\mu\sigma_\star}{L}\right)^{2/3}, \quad \eta \propto \textit{Const}, \quad \alpha \propto \textit{Const}.$$

Proof. Ignoring the constant terms, the iteration complexity that we obtain in Theorem 4 is

$$K = \widetilde{\mathcal{O}}\left(\max\left\{\frac{L}{\mu^2\varepsilon}, \frac{L\sigma^2}{\mu^4\varepsilon^3}, \frac{\sigma^2}{\varepsilon^2\mu^2}\right\}\right).$$

Since $\sigma = \sigma_\star/\sqrt{B}$, multiplying both sides of the above iteration complexity by B gives

$$T = \widetilde{\mathcal{O}}\left(\max\left\{\frac{LB}{\mu^2\varepsilon}, \frac{L\sigma_\star^2}{\mu^4\varepsilon^3}, \frac{\sigma_\star^2}{\varepsilon^2\mu^2}\right\}\right).$$

Since T is fixed, we cannot achieve an arbitrarily small accuracy. Instead, it is bounded as follows

$$\varepsilon = \widetilde{\mathcal{O}}\left(\max\left\{\frac{LB}{\mu^2 T}, \left(\frac{L\sigma_\star^2}{\mu^4 T}\right)^{1/3}, \frac{\sigma_\star}{\mu\sqrt{T}}\right\}\right).$$

Now we have two cases depending on T. If T is small enough, i.e., $T \leq \left(\frac{100\mu\sigma_\star}{\sqrt{30}L}\right)^2$, then we have

$$\left(\frac{1600\sigma_\star^2}{\mu^2 T}\right)^{1/2} \geq \left(\frac{320\sqrt{120}L\sigma_\star^2}{\mu^4 T}\right)^{1/3}.$$

For $T \geq \left(\frac{100\mu\sigma_\star}{\sqrt{30}L}\right)^2$, then we have

$$\left(\frac{1600\sigma_\star^2}{\mu^2 T}\right)^{1/2} \leq \left(\frac{320\sqrt{120}L\sigma_\star^2}{\mu^4 T}\right)^{1/3}.$$

In the large-T regime, we want to choose a batch size as large as possible so that we still have the minimum error $\varepsilon \sim T^{-1/3}$

$$\frac{LB}{\mu^2 T} = \left(\frac{L\sigma_\star^2}{\mu^4 T}\right)^{1/3} \Rightarrow B = T^{2/3} \cdot \left(\frac{\mu\sigma_\star}{L}\right)^{2/3}$$

In this regime, the choice of momentum parameter α is

$$\alpha \propto \frac{\mu^2 BS}{\sigma_\star^2}\left(\frac{\sigma_\star^2 L}{\mu^4 T}\right)^{2/3} = \frac{\mu^2}{\sigma_\star^2} \cdot T^{2/3} \cdot \left(\frac{\mu\sigma_\star}{L}\right)^{2/3}\left(\frac{\sigma_\star^2 L}{\mu^4 T}\right)^{2/3} = 1,$$

i.e., it should remain constant. The step size η should be chosen as

$$\eta = \min\left\{\frac{1}{L}, \frac{\alpha}{\sqrt{120}L}\right\} \propto \frac{1}{L},$$

and it is independent of the token budget. In other words, it also should remain constant. □

E CONVERGENCE ANALYSIS FOR ADAMW

Table E.1: Notation used in the analysis of the `AdamW` algorithm.

Notation	Meaning	Interpretation
P_k	$\operatorname{diag}\left(\frac{1}{\sqrt{v_k}+\nu}\right)$	AdamW's preconditioner
G	$\sup_{i,k}\lvert g_k^i\rvert$	Upper bound on the coordinate-wise stochastic gradient
p_l	$\frac{1}{G+\nu}$	Lower bound on AdamW's preconditioner
p_u	$\frac{1}{\nu}$	Upper bound on AdamW's preconditioner
r	$\min\left\{\frac{p_l\mu\varepsilon}{16}, \sqrt{\frac{p_l\alpha\varepsilon}{64\eta}}\right\}$	Scaling factor of weight decay λ with η and α
e_k	$\mathbb{E}[\lVert m_k - \nabla f(x_k)\rVert^2]$	Expected momentum error
F_k	$\mathbb{E}[f(x_k) - f^\star]$	Expected function sub-optimality
V_k	$\frac{\eta p_u}{\alpha} e_k$	Momentum error term in the Lyapunov function
Φ_k	$f(x_k) - f^\star + \frac{\eta p_u}{\alpha}\lVert m_k - \nabla f(x_k)\rVert^2$	Lyapunov function
Y_k	$(\mathbb{E}[\Phi_k] - \varepsilon/2)_+$	Auxilary
κ_ε	$\min\left\{\frac{\alpha}{4}, \frac{\eta p_l \mu^2 \varepsilon}{16}\right\}$	Contraction decay factor of the Lyapunov function

Algorithm 2 `AdamW`

Input: Initial point $x_0 \neq 0$, hyperparameters $\eta > 0$, $\alpha, \beta \in (0, 1]$, $\lambda \geq 0$, $\nu > 0$, momentum buffers $m_{-1}, v_{-1} = 0$
Initialize $m_{-1} = 0$, $v_{-1} = 0$
for $k = 0, 1, \ldots, K-1$ **do**
  Compute stochastic gradient g_k at x_k
  Update first moment: $m_k = (1-\alpha)m_{k-1} + \alpha g_k$
  Update second moment: $v_k = (1-\beta)v_{k-1} + \beta(g_k \odot g_k)$
  Set preconditioner: $P_k = \operatorname{diag}\left(1/(\sqrt{v_k} + \nu)\right)$
  Update parameters: $x_{k+1} = (1-\eta\lambda)x_k - \eta P_k m_k$
end for

Our analysis assumes that Assumptions 1 and 2 hold on a compact set containing the entire `AdamW` trajectory.

E.1 AUXILIARY LEMMAS

Lemma 5. *Let Assumption 4 hold. Then, for the second-order momentum buffer v_k and the preconditioner $P_k \coloneqq \operatorname{diag}(1/(\sqrt{v_k} + \nu))$, we have*

$$0 \leq v_k^i \leq G^2, \quad \frac{1}{G+\nu} \leq P_k^i = \frac{1}{\sqrt{v_k^i} + \nu} \leq \frac{1}{\nu}, \quad i \in [d], k \geq 0,$$

i.e., the inequality $p_l I \preceq P_k \preceq p_u I$ holds with $p_l \coloneqq \frac{1}{G+\nu}, p_u \coloneqq \frac{1}{\nu}$, where I is the identity matrix. In particular, this implies that

$$p_l\|u\|_2^2 \leq \langle P_k u, u\rangle, \quad \|P_k u\|_2 \leq p_u\|u\|_2$$

for any vector u.

Proof. Recall that $v_{k+1} = (1-\beta)v_k + \beta(g_{k+1} \odot g_{k+1})$ with $v_{-1} = 0$. From Assumption 4, we have $0 \leq |g_k^i| \leq G$ for all $i \in [d]$. We first prove that $0 \leq v_k^i \leq G^2$ by induction. The base case is $k = -1$, where $v_{-1}^i = 0$. Assume that $0 \leq v_k^i \leq G^2$ at iteration k. Then at iteration $k+1$, we have

$$v_{k+1}^i = (1-\beta)v_k^i + \beta|g_{k+1}^i|^2 \leq (1-\beta)G^2 + \beta G^2 \leq G^2,$$

where we use the induction hypothesis $v_k^i \le G^2$ and Assumption 4 $|g_{k+1}^i|^2 \le G^2$. The case $v_{k+1}^i \ge 0$ is straightforward.

Since $P_k^i = \frac{1}{\sqrt{v_k^i}+\nu}$, then $\frac{1}{\nu} \ge P_k^i \ge \frac{1}{G+\nu}$ for all $k \ge 0$ and $i \in [d]$. This implies that

$$P_k - p_l I = \operatorname{diag}(P_k^1 - p_l, \dots, P_k^d - p_l) \succeq 0, \quad p_u I - P_k = \operatorname{diag}(p_u - P_k^1, \dots, p_u - P_k^d) \succeq 0,$$

which concludes the proof. □

The following self-contained, simplified lemma from Li et al. (2025b); Xie & Li (2024) is essential to the proof.

Lemma 6. *Let the momentum parameters in* `AdamW` *satisfy*

$$\frac{\alpha}{4} \le \beta \le \alpha < 1, \quad \textit{or} \quad \beta = \alpha = 1,$$

and

$$\lambda > 0, \quad \eta \ge 0, \quad 0 \le \eta\lambda \le 1, \quad \nu > 0, \quad m_{-1} = 0, \quad v_{-1} = 0.$$

Then, for all $k \ge 0$ and $i \in [d]$,

$$|m_k^i|^2 \le 4v_k^i.$$

In particular, if $v_k^i > 0$, then

$$\frac{|m_k^i|^2}{v_k^i} \le 4.$$

Moreover, for every $k \ge 0$,

$$\|x_k\|_\infty - \frac{2}{\lambda} \le (1-\eta\lambda)^k \left(\|x_0\|_\infty - \frac{2}{\lambda} \right).$$

Proof. Let us consider the case $\alpha = \beta = 1$ first. This implies that

$$m_k = g_k, \quad v_k = g_k^{\odot 2}, \quad \forall k \ge 0.$$

Therefore

$$|m_k^i|^2 = |g_k^i|^2 = v_k^i \le 4v_k^i.$$

Thus, it remains to consider the case $\alpha/4 \le \beta \le \alpha < 1$. By expanding the recursions from $m_{-1} = 0$ and $v_{-1} = 0$, we obtain

$$m_k^i = \alpha \sum_{j=0}^{k} (1-\alpha)^{k-j} g_j^i, \quad v_k^i = \beta \sum_{j=0}^{k} (1-\beta)^{k-j} |g_j^i|^2.$$

By Jensen's and Cauchy–Schwarz inequalities,

$$\begin{aligned}
|m_k^i|^2 &= \left(\alpha \sum_{j=0}^{k} (1-\alpha)^{k-j} g_j^i \right)^2 \\
&= \left(\sum_{j=0}^{k} (1-\beta)^{(k-j)/2} |g_j^i| \cdot \frac{\alpha(1-\alpha)^{k-j}}{(1-\beta)^{(k-j)/2}} \right)^2 \\
&\le \left(\sum_{j=0}^{k} (1-\beta)^{k-j} |g_j^i|^2 \right) \left(\alpha^2 \sum_{j=0}^{k} \left(\frac{(1-\alpha)^2}{1-\beta} \right)^{k-j} \right) \\
&= \frac{v_k^i}{\beta} \cdot \alpha^2 \sum_{j=0}^{k} \left(\frac{(1-\alpha)^2}{1-\beta} \right)^{k-j} \\
&\le \frac{v_k^i}{\beta} \alpha^2 \sum_{j=0}^{k} (1-\alpha)^{k-j} \\
&\le \frac{v_k^i}{\beta} \alpha^2 \frac{1}{\alpha} \\
&\le v_k^i \frac{\alpha}{\beta},
\end{aligned}$$

where we use the fact that $1-\alpha \le 1-\beta$. Thus for $\beta \ge \alpha/4$, we have:

$$|m_k^i|^2 \le \frac{v_k^i \alpha}{\beta} \le 4v_k^i,$$

where the last step uses $\alpha/4 < \beta \le 1$. This proves the first claim of the lemma.

For the second claim, the first part implies that, for every coordinate i,

$$\frac{|m_k^i|}{\sqrt{v_k^i}+\nu} \le \frac{\sqrt{4v_k^i}}{\sqrt{v_k^i}+\nu} \le 2.$$

Therefore

$$\left\| \frac{\eta m_k}{\sqrt{v_k}+\nu} \right\|_\infty = \max_{i\in[d]} \left| \frac{\eta m_k^i}{\sqrt{v_k^i}+\nu} \right| \le 2\eta.$$

By assumption of the lemma, we have $0 \le 1-\eta\lambda \le 1$. Using the update rule for x_{k+1} and the triangle inequality,

$$\begin{aligned}
\|x_{k+1}\|_\infty - \frac{2}{\lambda} &= \|(1-\eta\lambda)x_k - \eta P_k m_k\|_\infty - \frac{2}{\lambda} \le (1-\eta\lambda)\|x_k\|_\infty + \|\eta P_k m_k\|_\infty - \frac{2}{\lambda} \\
&= (1-\eta\lambda)\|x_k\|_\infty + \left\| \frac{\eta m_k}{\sqrt{v_k}+\nu} \right\|_\infty - \frac{2}{\lambda} \\
&\le (1-\eta\lambda)\|x_k\|_\infty + 2\eta - \frac{2}{\lambda} \\
&= (1-\eta\lambda)\left(\|x_k\|_\infty - \frac{2}{\lambda} \right).
\end{aligned}$$

Doing the same inequality $k+1$ times, this recursion gives

$$\|x_{k+1}\|_\infty - \frac{2}{\lambda} \le (1-\eta\lambda)^{k+1}\left(\|x_0\|_\infty - \frac{2}{\lambda} \right),$$

which proves the second statement of the lemma. □

From Lemma 5, we have that $p_l I \preceq P_s \preceq p_u I$ for all iterations s. We use this fact in the next lemma.

Lemma 7. *Let $0 \le \eta\lambda \le 1$, $0 < \varepsilon \le 1$, $0 < \alpha \le 1$. We define*

$$r \coloneqq \min\left\{ \frac{p_l \mu \varepsilon}{16}, \sqrt{\frac{p_l \alpha \varepsilon}{64\eta}} \right\}, \quad \kappa_\varepsilon \coloneqq \min\left\{ \frac{\alpha}{4}, \frac{\eta p_l \mu^2 \varepsilon}{16} \right\}.$$

Assume that $1 \le k \le K$, $\kappa_\varepsilon K \le 3$, and hyperparameters η and α are chosen such that

$$\frac{\kappa_\varepsilon}{\varepsilon}\left(\mathbb{E}[\Phi_0]+1\right) \le \frac{1}{4}.$$

If for all $0 \le s < k$, we have

$$\mathbb{E}[\Phi_{s+1}] \le \mathbb{E}[\Phi_s] - \frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle] + \frac{\kappa_\varepsilon \varepsilon}{4}, \quad \textit{and} \tag{9}$$

$$\mathbb{E}[\Phi_s] \le \frac{\varepsilon}{2} + \frac{\mathbb{E}[\Phi_0]+1}{1+\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0]+1)s}, \tag{10}$$

then

$$\left(\mathbb{E}\|x_k\|_2^2\right)^{1/2} \le \|x_0\|_2 + \sqrt{\frac{40\eta p_u \varepsilon}{\kappa_\varepsilon}}\left(1+\lceil \log_2 K \rceil\right). \tag{11}$$

Consequently, the choice

$$\lambda \le \frac{r}{\|x_0\|_2 + \sqrt{40\eta p_u \varepsilon/\kappa_\varepsilon}(1+\lceil \log_2 K \rceil)} \tag{12}$$

ensures $\lambda^2 \mathbb{E}[\|x_k\|_2^2] \le r^2$ at index k.

Proof. Summing (9) for all $0 \le i < s \le k$ and using $\Phi_s \ge 0$ gives

$$\sum_{t=i}^{s-1} \eta \mathbb{E}[\langle m_t, P_t m_t \rangle] \le 8 \left(\mathbb{E}[\Phi_i] + (s-i)\frac{\kappa_\varepsilon \varepsilon}{4} \right). \tag{13}$$

Since $P_t^2 \preceq p_u P_t$ by Lemma 5. Using the Cauchy–Schwarz inequality yields

$$\begin{aligned}
\sum_{t=i}^{s-1} \eta \left(\mathbb{E}[\|P_t m_t\|_2^2] \right)^{1/2} &\le \left(\sum_{t=i}^{s-1} \eta \right)^{1/2} \times \left(\sum_{t=i}^{s-1} \eta \mathbb{E}[\|P_t m_t\|_2^2] \right)^{1/2} \\
&\le \left((s-i)\eta \right)^{1/2} \times \left(p_u \sum_{t=i}^{s-1} \eta \mathbb{E}[\langle m_t, P_t m_t \rangle] \right)^{1/2} \\
&= \left(\eta(s-i) p_u \sum_{t=i}^{s-1} \eta \mathbb{E}[\langle m_t, P_t m_t \rangle] \right)^{1/2} \\
&\le \left(8\eta p_u \left((s-i)\mathbb{E}[\Phi_i] + (s-i)^2 \frac{\kappa_\varepsilon \varepsilon}{4} \right) \right)^{1/2},
\end{aligned} \tag{14}$$

where the last inequality is due to (13). Note that partitioning the set $\{0, \dots, k-1\}$ into intervals of the form: $\{0\}, [1,2), [2,4), [4,8), \dots$ and truncating the last interval, gives us at most $1 + \lceil \log_2 K \rceil$ nonempty intervals. By the construction, for an interval starting at $i \ge 1$, its length satisfies $s - i \le i$. Now, using (10) and $K\kappa_\varepsilon \le 3$, we obtain

$$\begin{aligned}
(s-i)\mathbb{E}[\Phi_i] &\le (s-i)\frac{\varepsilon}{2} + (s-i)\frac{\mathbb{E}[\Phi_0]+1}{1 + \frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0]+1)i} \\
&\le \frac{K\varepsilon}{2} + \frac{(s-i)(\mathbb{E}[\Phi_0]+1)}{\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0]+1)i} \\
&\le \frac{3\varepsilon}{2\kappa_\varepsilon} + \frac{\varepsilon}{\kappa_\varepsilon} \cdot \frac{s-i}{i} \le \frac{5\varepsilon}{2\kappa_\varepsilon},
\end{aligned}$$

and

$$(s-i)^2 \frac{\kappa_\varepsilon \varepsilon}{4} \le K^2 \frac{\kappa_\varepsilon \varepsilon}{4} \le \frac{9}{\kappa_\varepsilon^2} \cdot \frac{\kappa_\varepsilon \varepsilon}{4} = \frac{9\varepsilon}{4\kappa_\varepsilon}.$$

For the first singleton $\{0\}$, we have $i = 0, s = 1$, and we instead use the fact that $\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0]+1) \le \frac{1}{4}$. It implies that

$$(s-i)\mathbb{E}[\Phi_i] = \mathbb{E}[\Phi_0] \le \frac{\varepsilon}{4\kappa_\varepsilon} \le \frac{5\varepsilon}{2\kappa_\varepsilon}.$$

We combine the obtained inequalities to obtain

$$\sum_{t=i}^{s-1} \eta \left(\mathbb{E}[\|P_t m_t\|_2^2] \right)^{1/2} \le \left(8\eta p_u \left(\frac{5\varepsilon}{2\kappa_\varepsilon} + \frac{9\varepsilon}{4\kappa_\varepsilon} \right) \right)^{1/2} = \sqrt{\frac{38\eta p_u \varepsilon}{\kappa_\varepsilon}} \le \sqrt{\frac{40\eta p_u \varepsilon}{\kappa_\varepsilon}}. \tag{15}$$

Finally, combining the update rule, Minkowski's inequality, and the choice $0 \le 1 - \eta\lambda \le 1$ imply that

$$\begin{aligned}
\left(\mathbb{E}\|x_{s+1}\|_2^2 \right)^{1/2} &= \left(\mathbb{E}\|(1-\lambda\eta)x_s - \eta P_s m_s\|_2^2 \right)^{1/2} \\
&\le \left(\mathbb{E}[\|(1-\lambda\eta)x_s\|_2^2] \right)^{1/2} + \left(\mathbb{E}[\|\eta P_s m_s\|_2^2] \right)^{1/2} \\
&\le (1-\eta\lambda) \left(\mathbb{E}\|x_s\|_2^2 \right)^{1/2} + \eta \left(\mathbb{E}\|P_s m_s\|_2^2 \right)^{1/2} \\
&\le \left(\mathbb{E}\|x_s\|_2^2 \right)^{1/2} + \eta \left(\mathbb{E}\|P_s m_s\|_2^2 \right)^{1/2}.
\end{aligned}$$

We sum over the intervals, using (15), and obtain (11). Note that the case $k = 0$ trivially holds. Multiplying by λ and choosing λ such that (12) holds, we prove the final claim. □

Lemma 8. *Let Assumption 1 hold and $p_l I \preceq P_k \preceq p_u I$. If $\eta \leq 1/(4Lp_u)$, then*

$$f(x_{k+1}) \leq f(x_k) - \frac{\eta p_l}{4}\|\nabla f(x_k)\|_2^2 - \frac{\eta}{4}\langle m_k, P_k m_k\rangle + \frac{\eta p_u}{2}\|m_k - \nabla f(x_k)\|_2^2 + \eta(1/p_l + L\eta)\lambda^2\|x_k\|_2^2.$$

Proof. Using Assumption 1 and the update rule $x_{k+1} - x_k = -\eta(P_k m_k + \lambda x_k)$, we have

$$f(x_{k+1}) \leq f(x_k) - \eta\langle\nabla f(x_k), P_k m_k\rangle - \eta\lambda\langle\nabla f(x_k), x_k\rangle + \frac{L\eta^2}{2}\left\|P_k m_k + \lambda x_k\right\|^2. \tag{16}$$

Since $P_k \preceq p_u I$ by Lemma 5, then we also have $P_k^2 \preceq p_u P_k$. Thus, $\|P_k m_k\|_2^2 \leq p_u\langle m_k, P_k m_k\rangle$. Now we use Young's inequality and obtain

$$\|P_k m_k + \lambda x_k\|_2^2 \leq 2\|P_k m_k\|_2^2 + 2\lambda^2\|x_k\|_2^2 \leq 2p_u\langle m_k, P_k m_k\rangle + 2\lambda^2\|x_k\|_2^2. \tag{17}$$

Using the equality

$$\langle m_k - \nabla f(x_k),\, P_k(m_k - \nabla f(x_k))\rangle = \langle m_k, P_k m_k\rangle - 2\langle\nabla f(x_k), P_k m_k\rangle + \langle\nabla f(x_k), P_k\nabla f(x_k)\rangle,$$

we rewrite the term $-\langle\nabla f(x_k), P_k m_k\rangle$ as follows

$$-\langle\nabla f(x_k), P_k m_k\rangle = -\frac{1}{2}\langle\nabla f(x_k), P_k\nabla f(x_k)\rangle - \frac{1}{2}\langle m_k, P_k m_k\rangle + \frac{1}{2}\langle m_k - \nabla f(x_k), P_k(m_k - \nabla f(x_k))\rangle. \tag{18}$$

Substituting the equations (17) and (18) into the quadratic smoothness bound (16), we obtain

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) - \frac{\eta}{2}\langle\nabla f(x_k), P_k\nabla f(x_k)\rangle - \frac{\eta}{2}\langle m_k, P_k m_k\rangle \\ &\quad + \frac{\eta}{2}\langle m_k - \nabla f(x_k), P_k(m_k - \nabla f(x_k))\rangle \\ &\quad - \eta\lambda\langle\nabla f(x_k), x_k\rangle + L\eta^2 p_u\langle m_k, P_k m_k\rangle + L\eta^2\lambda^2\|x_k\|_2^2 \\ &= f(x_k) - \frac{\eta}{2}\langle\nabla f(x_k), P_k\nabla f(x_k)\rangle - \frac{\eta}{2}(1 - 2L\eta p_u)\langle m_k, P_k m_k\rangle \\ &\quad + \frac{\eta}{2}\langle m_k - \nabla f(x_k), P_k(m_k - \nabla f(x_k))\rangle - \eta\lambda\langle\nabla f(x_k), x_k\rangle \\ &\quad + L\eta^2\lambda^2\|x_k\|_2^2. \end{aligned}$$

Since $\eta \leq 1/(4Lp_u)$, we obtain

$$-\frac{\eta}{2}(1 - 2L\eta p_u)\langle m_k, P_k m_k\rangle \leq -\frac{\eta}{4}\langle m_k, P_k m_k\rangle.$$

Using $P_k \succeq p_l I$ and $P_k \preceq p_u I$ from Lemma 5, we get

$$f(x_{k+1}) \leq f(x_k) - \frac{\eta p_l}{2}\|\nabla f(x_k)\|_2^2 - \frac{\eta}{4}\langle m_k, P_k m_k\rangle + \frac{\eta p_u}{2}\|m_k - \nabla f(x_k)\|_2^2 - \eta\lambda\langle\nabla f(x_k), x_k\rangle + L\eta^2\lambda^2\|x_k\|_2^2. \tag{19}$$

Using Young's inequality, we bound the cross term as follows

$$-\eta\lambda\langle\nabla f(x_k), x_k\rangle \leq \frac{\eta p_l}{4}\|\nabla f(x_k)\|_2^2 + \frac{\eta\lambda^2}{p_l}\|x_k\|_2^2.$$

Substituting this bound into (19), we prove the statement of the lemma. □

Lemma 9. *Let Assumptions 1, 3, and 4 hold, and let $0 < \alpha \leq 1$. Then, for every $k \geq 0$,*

$$e_{k+1} \leq (1 - \alpha)e_k + \frac{2L^2\eta^2 p_u}{\alpha}\mathbb{E}\langle m_k, P_k m_k\rangle + \frac{2L^2\eta^2\lambda^2}{\alpha}\mathbb{E}[\|x_k\|_2^2] + \alpha^2\sigma^2,$$

where $e_k = \mathbb{E}[\|m_k - \nabla f(x_k)\|_2^2]$.

Proof. We can decompose the term $m_{k+1} - \nabla f(x_{k+1}) = A_k + B_k$ with

$$\begin{aligned} m_{k+1} - \nabla f(x_{k+1}) &= (1-\alpha)m_k + \alpha g_{k+1} - \nabla f(x_{k+1}) \\ &= (1-\alpha)(m_k - \nabla f(x_k)) + (1-\alpha)(\nabla f(x_k) - \nabla f(x_{k+1})) \\ &\quad + \alpha(g_{k+1} - \nabla f(x_{k+1})), \end{aligned}$$

i.e., $m_{k+1} - \nabla f(x_{k+1}) = A_k + B_k$ with

$$\begin{aligned} A_k &\coloneqq (1-\alpha)(m_k - \nabla f(x_k)) + (1-\alpha)(\nabla f(x_k) - \nabla f(x_{k+1})), \\ B_k &\coloneqq \alpha(g_{k+1} - \nabla f(x_{k+1})). \end{aligned}$$

Note that the random vectors A_k and x_{k+1} are determined by the history $g_0, \ldots, g_k$. Using the conditional unbiasedness and the variance bound, we obtain

$$\mathbb{E}[B_k \mid g_0, \ldots, g_k] = 0, \quad \mathbb{E}[\|B_k\|_2^2 \mid g_0, \ldots, g_k] \leq \alpha^2\sigma^2.$$

Consequently,

$$\mathbb{E}[\|A_k + B_k\|_2^2 \mid g_0, \ldots, g_k] = \|A_k\|_2^2 + \mathbb{E}[\|B_k\|_2^2 \mid g_0, \ldots, g_k].$$

Using Young's inequality, we get

$$\|A_k\|_2^2 \leq (1-\alpha)\|m_k - \nabla f(x_k)\|_2^2 + \frac{1}{\alpha}\|\nabla f(x_k) - \nabla f(x_{k+1})\|_2^2.$$

For $\alpha = 1$, we have $A_k = 0$, and this bound becomes trivial. Using Assumption 1, the update rule, and $P_k^2 \preceq p_u P_k$ from Lemma 5, we get

$$\begin{aligned} \|\nabla f(x_k) - \nabla f(x_{k+1})\|_2^2 &\leq L^2\eta^2\|P_k m_k + \lambda x_k\|_2^2 \\ &\leq 2L^2\eta^2\lambda^2\|x_k\|_2^2 + 2L^2\eta^2\|P_k m_k\|_2^2 \\ &\leq 2L^2\eta^2 p_u\langle m_k, P_k m_k\rangle + 2L^2\eta^2\lambda^2\|x_k\|_2^2. \end{aligned}$$

Taking the full expectation and collecting previous bounds, we get

$$e_{k+1} \leq (1-\alpha)e_k + \frac{2L^2\eta^2 p_u}{\alpha}\left(\mathbb{E}[\langle m_k, P_k m_k\rangle]\right) + \frac{2L^2\eta^2\lambda^2}{\alpha}\mathbb{E}[\|x_k\|_2^2] + \alpha^2\sigma^2,$$

proving the statement of the lemma.

□

The following Lyapunov proof for `AdamW` is based on a Lyapunov function, which is constructed similarly to that for `SGD` with momentum,

$$\Phi_k \coloneqq f(x_k) - f^\star + \frac{\eta p_u}{\alpha}\left\|m_k - \nabla f(x_k)\right\|,$$

Moreover, we introduce the notation $F_k = \mathbb{E}[f(x_k) - f^\star]$, $V_k = \frac{\eta p_u}{\alpha}e_k$ for shortness. This implies that $\mathbb{E}[\Phi_k] = F_k + V_k$.

Lemma 10. *Let Assumptions 1, 2, 3, and 4 hold. If* $0 < \alpha \leq 1$ *and* $\eta \leq \alpha/(4Lp_u)$*, then*

$$\begin{aligned} \mathbb{E}[\Phi_{k+1}] \leq \mathbb{E}[\Phi_k] - \frac{\eta p_l \mu^2}{4}F_k^2 - \frac{\eta}{8}\mathbb{E}[\langle m_k, P_k m_k\rangle] - \frac{\alpha}{2}V_k \\ + \eta p_u \alpha\sigma^2 + \frac{2\eta\lambda^2}{p_l}\mathbb{E}[\|x_k\|_2^2]. \end{aligned} \tag{20}$$

Proof. Take expectation in Lemma 8 and add Lemma 9 multiplied by $\eta p_u/\alpha$:

$$\begin{aligned} \mathbb{E}[\Phi_{k+1}] \leq \mathbb{E}[\Phi_k] - \frac{\eta p_l}{4}\mathbb{E}[\|\nabla f(x_k)\|_2^2] + \left(-\frac{\eta}{4} + \frac{2L^2\eta^3 p_u^2}{\alpha^2}\right)\mathbb{E}[\langle m_k, P_k m_k\rangle] \\ + \left(\frac{\eta p_u}{2} - \eta p_u\right)e_k + \eta p_u\alpha\sigma^2 \\ + \eta\left(\frac{1}{p_l} + L\eta + \frac{2L^2\eta^2 p_u}{\alpha^2}\right)\lambda^2\mathbb{E}[\|x_k\|_2^2]. \end{aligned} \tag{21}$$

The step-size restriction implies

$$\frac{2L^2\eta^3 p_u^2}{\alpha^2} \leq \frac{\eta}{8}, \quad \left(\frac{\eta p_u}{2} - \eta p_u\right) e_k = -\frac{\alpha}{2} V_k. \tag{22}$$

Moreover, since $\alpha \leq 1$ and $p_l \leq p_u$,

$$\frac{1}{p_l} + L\eta + \frac{2L^2\eta^2 p_u}{\alpha^2} \leq \frac{1}{p_l} + \frac{1}{4p_u} + \frac{1}{8p_u} \leq \frac{2}{p_l}. \tag{23}$$

Using (22) and (23) into (21), we obtain

$$\begin{aligned}\mathbb{E}[\Phi_{k+1}] \leq \mathbb{E}[\Phi_k] - \frac{\eta p_l}{4}\mathbb{E}\|\nabla f(x_k)\|_2^2 - \frac{\eta}{8}\mathbb{E}[\langle m_k, P_k m_k\rangle] - \frac{\alpha}{2}V_k \\ + \eta p_u \alpha\sigma^2 + \frac{2\eta\lambda^2}{p_l}\mathbb{E}[\|x_k\|_2^2].\end{aligned} \tag{24}$$

It remains to use Assumption 2 and Jensen's inequality, that imply

$$\mathbb{E}[\|\nabla f(x_k)\|_2^2] \geq \mu^2 \mathbb{E}[(f(x_k) - f^\star)^2] \geq \mu^2 F_k^2. \tag{25}$$

Using (25) in (24), we obtain the statement of the lemma. □

Lemma 11. *For $0 < \varepsilon \leq 1$ and $0 < \alpha \leq 1$, define*

$$\kappa_\varepsilon := \min\left\{\frac{\alpha}{4}, \frac{\eta p_l \mu^2 \varepsilon}{16}\right\}.$$

Then, for every $k \geq 0$, we have

$$\frac{\eta p_l \mu^2}{8} F_k^2 + \frac{\alpha}{4} V_k \geq \kappa_\varepsilon \left(\mathbb{E}[\Phi_k] - \varepsilon/2\right)_+.$$

Moreover, if

$$\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1) \leq \frac{\alpha}{4}, \quad \mathbb{E}[\Phi_k] \leq \mathbb{E}[\Phi_0] + 1, \tag{26}$$

then the stronger quadratic bound holds:

$$\frac{\eta p_l \mu^2}{4} F_k^2 + \frac{\alpha}{2} V_k \geq \frac{\kappa_\varepsilon}{\varepsilon}\left(\mathbb{E}[\Phi_k]\right)^2. \tag{27}$$

If these conditions hold for all iterations up to k and

$$\mathbb{E}[\Phi_{k+1}] \leq \mathbb{E}[\Phi_k] - \frac{\kappa_\varepsilon}{\varepsilon}\left(\mathbb{E}[\Phi_k]\right)^2 + \frac{\kappa_\varepsilon \varepsilon}{4}, \quad 0 \leq s < k \tag{28}$$

then

$$Y_{k+1} \leq Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2, \quad Y_k \leq \frac{\mathbb{E}[\Phi_0] + 1}{1 + \frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1)k}. \tag{29}$$

Proof. Let $Y_k := (\mathbb{E}[\Phi_k] - \varepsilon/2)_+ = (F_k + V_k - \varepsilon/2)_+$. To study the convergence of Y_k, we consider the following two cases.

Case 1: $F_k \geq \varepsilon/2$**.** Then $F_k^2 \geq \frac{\varepsilon}{2} F_k$, hence

$$\frac{\eta p_l \mu^2}{8} F_k^2 \geq \frac{\eta p_l \mu^2 \varepsilon}{16} F_k \geq \kappa_\varepsilon F_k.$$

Besides, $\frac{\alpha}{4} V_k \geq \kappa_\varepsilon V_k$. Summing these two inequalities, we obtain

$$\frac{\eta p_l \mu^2}{8} F_k^2 + \frac{\alpha}{4} V_k \geq \kappa_\varepsilon (F_k + V_k) \geq \kappa_\varepsilon Y_k.$$

Case 2: $F_k \le \varepsilon/2$. Then $Y_k = (F_k + V_k - \varepsilon/2)_+ \le V_k$, so

$$\frac{\eta p_l \mu^2}{8} F_k^2 + \tfrac{\alpha}{4} V_k \ge \frac{\alpha}{4} V_k \ge \kappa_\varepsilon V_k \ge \kappa_\varepsilon Y_k.$$

We obtain that in both cases, the statement of the lemma holds.

For the quadratic bound, use $\mathbb{E}[\Phi_k] = F_k + V_k$ and $0 \le V_k \le \mathbb{E}[\Phi_0] + 1$ to obtain

$$(\mathbb{E}[\Phi_k])^2 = (F_k + V_k)^2 \le 2F_k^2 + 2V_k^2 \le 2F_k^2 + 2(\mathbb{E}[\Phi_0] + 1)V_k.$$

Multiplying by $\kappa_\varepsilon/\varepsilon$, we get

$$\frac{\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_k])^2 \le \frac{2\kappa_\varepsilon}{\varepsilon} F_k^2 + \frac{2\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_0] + 1) V_k$$

Since $\kappa_\varepsilon/\varepsilon \le \eta p_l \mu^2/16$ and $\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1) \le \frac{\alpha}{4}$, then

$$\frac{\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_k])^2 \le \frac{\eta p_l \mu^2}{8} F_k^2 + \frac{\alpha}{2} V_k \le \frac{\eta p_l \mu^2}{4} F_k^2 + \frac{\alpha}{2} V_k,$$

which proves (27).

Suppose that $\mathbb{E}[\Phi_k] \le \varepsilon/2$. Starting from (28) and subtracting $\varepsilon/2$, we obtain

$$\begin{aligned}
\mathbb{E}[\Phi_{k+1}] - \frac{\varepsilon}{2} &\le \mathbb{E}[\Phi_k] - \frac{\varepsilon}{2} - \frac{\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_k])^2 + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&= \mathbb{E}[\Phi_k] - \frac{\varepsilon}{2} - \frac{\kappa_\varepsilon}{\varepsilon} \left[(\mathbb{E}[\Phi_k])^2 - \left(\frac{\varepsilon}{2}\right)^2 \right] \\
&= \left(\mathbb{E}[\Phi_k] - \frac{\varepsilon}{2} \right) \left[1 - \frac{\kappa_\varepsilon}{\varepsilon} \left(\mathbb{E}[\Phi_k] + \frac{\varepsilon}{2} \right) \right].
\end{aligned}$$

The first factor is non-positive by the case assumption. To bound the second factor, we observe that

$$\mathbb{E}[\Phi_k] + \frac{\varepsilon}{2} \le 2 \left(\mathbb{E}[\Phi_0] + 1 \right),$$

because $\mathbb{E}[\Phi_k] \le \varepsilon/2 \le \mathbb{E}[\Phi_0] + 1$. Consequently,

$$1 - \frac{\kappa_\varepsilon}{\varepsilon} \left(\mathbb{E}[\Phi_k] + \frac{\varepsilon}{2} \right) \ge 1 - 2\frac{\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_0] + 1) \ge 1 - \frac{\alpha}{2} \ge \frac{1}{2},$$

where we used $(\kappa_\varepsilon/\varepsilon)(\mathbb{E}[\Phi_0] + 1) \le \alpha/4$ and $\alpha \le 1$. Thus, the product of the two factors is non-positive, implying

$$\mathbb{E}[\Phi_{k+1}] \le \frac{\varepsilon}{2}.$$

Taking the positive part therefore gives

$$Y_{k+1} = \left(\mathbb{E}[\Phi_{k+1}] - \frac{\varepsilon}{2} \right)_+ = 0.$$

Now suppose that $\mathbb{E}[\Phi_k] > \varepsilon/2$. By the definition of the positive part, we have

$$Y_k = \mathbb{E}[\Phi_k] - \frac{\varepsilon}{2} > 0, \quad \mathbb{E}[\Phi_k] = Y_k + \frac{\varepsilon}{2}.$$

Substituting this identity into (28) and subtracting $\varepsilon/2$ gives

$$\begin{aligned}
\mathbb{E}[\Phi_{k+1}] - \frac{\varepsilon}{2} &\le Y_k - \frac{\kappa_\varepsilon}{\varepsilon} \left(Y_k + \frac{\varepsilon}{2} \right)^2 + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&= Y_k - \frac{\kappa_\varepsilon}{\varepsilon} \left(Y_k^2 + \varepsilon Y_k + \frac{\varepsilon^2}{4} \right) + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&= Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2 - \kappa_\varepsilon Y_k \\
&\le Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2,
\end{aligned}$$

where we drop the non-positive term $-\kappa_\varepsilon Y_k$. To take positive parts, we verify that the final upper bound is nonnegative. Since

$$0 < Y_k \le \mathbb{E}[\Phi_k] \le \mathbb{E}[\Phi_0] + 1,$$

the hypothesis of the lemma implies

$$\frac{\kappa_\varepsilon}{\varepsilon} Y_k \le \frac{\kappa_\varepsilon}{\varepsilon} \left(\mathbb{E}[\Phi_0] + 1\right) \le \frac{\alpha}{4} \le \frac{1}{4}.$$

Consequently,

$$Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2 = Y_k \left(1 - \frac{\kappa_\varepsilon}{\varepsilon} Y_k\right) \ge \frac{3}{4} Y_k > 0.$$

By monotonicity of the positive-part operation,

$$\begin{aligned} Y_{k+1} &= \left(\mathbb{E}[\Phi_{k+1}] - \frac{\varepsilon}{2}\right)_+ \\ &\le \left(Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2\right)_+ \\ &= Y_k - \frac{\kappa_\varepsilon}{\varepsilon} Y_k^2. \end{aligned}$$

Together with the preceding case $Y_k = 0$, this proves the first inequality in (29).

We next solve this recursion by taking reciprocals. Whenever $Y_{k+1} > 0$, both sides of

$$0 < Y_{k+1} \le Y_k \left(1 - \frac{\kappa_\varepsilon}{\varepsilon} Y_k\right)$$

are positive. Taking the inverse of both sides, it reverses the inequality:

$$\begin{aligned} \frac{1}{Y_{k+1}} &\ge \frac{1}{Y_k \left(1 - \frac{\kappa_\varepsilon}{\varepsilon} Y_k\right)} \\ &= \frac{1}{Y_k} + \frac{\kappa_\varepsilon/\varepsilon}{1 - \frac{\kappa_\varepsilon}{\varepsilon} Y_k} \\ &\ge \frac{1}{Y_k} + \frac{\kappa_\varepsilon}{\varepsilon}, \end{aligned}$$

where he last step uses $0 < 1 - (\kappa_\varepsilon/\varepsilon) Y_k \le 1$.

If $Y_k > 0$, then all preceding values $Y_0, \ldots, Y_k$ are positive: the preceding case shows that a zero value cannot be followed by a positive value while the recursion holds. Thus, summing the inversed inequality from $t = 0$ to $k - 1$ yields

$$\frac{1}{Y_k} \ge \frac{1}{Y_0} + \frac{\kappa_\varepsilon}{\varepsilon} k \ge \frac{1}{\mathbb{E}[\Phi_0] + 1} + \frac{\kappa_\varepsilon}{\varepsilon} k,$$

where the last inequality uses $Y_0 \le \mathbb{E}[\Phi_0] + 1$. Taking the inverse once again gives

$$Y_k \le \frac{\mathbb{E}[\Phi_0] + 1}{1 + \frac{\kappa_\varepsilon}{\varepsilon} (\mathbb{E}[\Phi_0] + 1) k}.$$

If $Y_k = 0$, this bound holds immediately. Moreover, all subsequent values on the prefix where the recursion holds remain zero. This proves the second inequality in (29).

□

E.2 Proof of the Main Theorem

In the remaining proof, we assume that the initial point x_0 is fixed.

Theorem 5. *Let Assumptions 1, 2, 3, and 4 hold. Let $0 < \varepsilon < 1$ be the target accuracy. Set the hyperparameters of `AdamW` (Algorithm 2) are set such that*

$$\alpha := \min\left\{1, \frac{p_l \mu^2 \varepsilon^2}{128 p_u \sigma^2}\right\}, \quad m_{-1} = v_{-1} = 0,$$

and

$$\eta = \min\left\{\frac{\sqrt{p_l}\alpha}{4\sqrt{2} p_u^{3/2} (2L + \mu^2)}, \frac{\varepsilon}{32 p_u \sigma^2}\right\}. \tag{30}$$

Moreover, we require α and β to satisfy one of the two restrictions

$$\frac{\alpha}{4} \le \beta \le \alpha < 1, \quad \textit{or} \quad \alpha = \beta = 1.$$

Define

$$\kappa_\varepsilon := \min\left\{\frac{\alpha}{4}, \frac{\eta p_l \mu^2 \varepsilon}{16}\right\}, \quad r := \min\left\{\frac{p_l \mu \varepsilon}{16}, \sqrt{\frac{p_l \alpha \varepsilon}{64\eta}}\right\}.$$

and set the number of iterations and the weight decay as

$$K = \left\lceil \frac{2}{\kappa_\varepsilon} \right\rceil, \quad 0 \le \lambda \le \min\left\{\frac{1}{2\eta}, \frac{r}{\sqrt{d}(\|x_0\|_\infty + \sqrt{40\eta p_u \varepsilon/\kappa_\varepsilon}(1 + \lceil \log_2 K \rceil))}\right\}. \tag{31}$$

Then, for $0 \le k \le K$, we have

$$Y_k \le \frac{\mathbb{E}[\Phi_0] + 1}{1 + \frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1)k}, \quad \left(\mathbb{E}[\|x_k\|_2]^2\right)^{1/2} \le \|x_0\|_2 + \sqrt{\frac{40\eta p_u \varepsilon}{\kappa_\varepsilon}}(1 + \lceil \log_2 K \rceil), \tag{32}$$

and $\mathbb{E}[f(x_K) - f^\star] \le \varepsilon$.

Proof. First, the learning rate choice implies that the following inequalities are satisfied

$$\eta \le \frac{\alpha}{4Lp_u}, \frac{\sqrt{p_l}\alpha}{4\sqrt{2}Lp_u^{3/2}}, \frac{2\alpha}{p_l(2L + \mu^2)}, \quad \eta p_l(2L + \mu^2) \le 2\alpha,$$

since $2L + \mu^2 \ge L$ and $p_l \le p_u$. Conditional unbiasedness of the stochastic gradients implies that $\nabla f(x_0) = \mathbb{E}[g_0]$. The initialization $m_{-1} = 0$ gives $m_0 = \alpha g_0$. Therefore,

$$m_0 - \nabla f(x_0) = -(1 - \alpha)\nabla f(x_0) + \alpha(g_0 - \nabla f(x_0)).$$

Squaring and taking expectations, the cross term vanishes by conditional unbiasedness. Using Assumption 3 yields

$$\begin{aligned} e_0 &= \mathbb{E}[\|m_0 - \nabla f(x_0)\|_2^2] \\ &= (1 - \alpha)^2 \|\nabla f(x_0)\|_2^2 + \alpha^2 \mathbb{E}[\|g_0 - \nabla f(x_0)\|_2^2] \\ &\le (1 - \alpha)^2 \|\nabla f(x_0)\|_2^2 + \alpha^2 \sigma^2. \end{aligned}$$

By Assumption 1 and the lower bound $f^\star$, we have

$$f^\star \le f\left(x_0 - \frac{1}{L}\nabla f(x_0)\right) \le f(x_0) - \frac{1}{2L}\|\nabla f(x_0)\|_2^2.$$

Thus $\|\nabla f(x_0)\|_2^2 \le 2LF_0$. Combining this with Assumption 2 gives

$$\mu^2 F_0^2 \le \|\nabla f(x_0)\|_2^2 \le 2LF_0.$$

If $F_0 > 0$, dividing by F_0 yields the bound below; if $F_0 = 0$, it holds immediately. Hence

$$0 \le F_0 \le \frac{2L}{\mu^2}. \tag{33}$$

The initial momentum error therefore satisfies

$$e_0 \le 2L(1 - \alpha)^2 F_0 + \alpha^2 \sigma^2.$$

Remind that the learning rate restriction implies that

$$\eta \le \frac{\alpha}{4Lp_u}.$$

Consequently, using also $\eta p_u \sigma^2 \le \varepsilon/32$, $0 < \alpha \le 1$, and $0 < \varepsilon \le 1$, we obtain

$$\begin{aligned} \mathbb{E}[\Phi_0] &= F_0 + \frac{\eta p_u}{\alpha} e_0 \\ &\le F_0 + \frac{2L\eta p_u}{\alpha}(1 - \alpha)^2 F_0 + \eta p_u \alpha \sigma^2 \\ &\le F_0 + \frac{(1 - \alpha)^2}{2} F_0 + \frac{\alpha\varepsilon}{32} \le \frac{3}{2}F_0 + \frac{1}{32}. \end{aligned} \tag{34}$$

In particular,

$$\mathbb{E}[\Phi_0] + 1 \le \frac{3}{2}F_0 + \frac{33}{32} \le 2(F_0 + 1).$$

We now check where the minimum is attained in the definitions of κ_ε and r. The learning rate restriction also implies that

$$\eta \le \frac{2\alpha}{p_l(2L + \mu^2)} \le \frac{2\alpha}{p_l\mu^2(F_0 + 1)}.$$

Here the second inequality follows from (33). Since $0 < \varepsilon \le 1$ and $F_0 + 1 \ge 1$, we obtain that

$$\eta p_l \mu^2 \varepsilon \le \frac{2\alpha\varepsilon}{F_0 + 1} \le 4\alpha.$$

Dividing by 16 shows that

$$\frac{\eta p_l \mu^2 \varepsilon}{16} \le \frac{\alpha}{4}.$$

Thus, the second term attains the minimum in the definition of κ_ε, and

$$\kappa_\varepsilon = \min\left\{\frac{\alpha}{4}, \frac{\eta p_l \mu^2 \varepsilon}{16}\right\} = \frac{\eta p_l \mu^2 \varepsilon}{16}. \tag{35}$$

The same inequality identifies the active branch of r. Indeed, since all quantities are positive, we have that

$$\left(\frac{p_l\mu\varepsilon}{16}\right)^2 \le \frac{p_l\alpha\varepsilon}{64\eta} \quad \iff \quad \eta p_l \mu^2 \varepsilon \le 4\alpha.$$

Taking square roots therefore gives

$$r = \min\left\{\frac{p_l\mu\varepsilon}{16}, \sqrt{\frac{p_l\alpha\varepsilon}{64\eta}}\right\} = \frac{p_l\mu\varepsilon}{16}.$$

Combining the initial Lyapunov bound (34) with the identity for κ_ε in (35) and learning rate restriction (30), we obtain

$$\frac{\kappa_\varepsilon}{\varepsilon}\left(\mathbb{E}[\Phi_0] + 1\right) = \frac{\eta p_l \mu^2}{16}\left(\mathbb{E}[\Phi_0] + 1\right) \le \frac{\eta p_l \mu^2}{16} 2(F_0 + 1) \le \frac{\eta p_l (2L + \mu^2)}{8} \le \frac{\alpha}{4} \le \frac{1}{4}.$$

Finally, since $K = \lceil 2/\kappa_\varepsilon \rceil$,

$$\frac{2}{\kappa_\varepsilon} \le K < \frac{2}{\kappa_\varepsilon} + 1.$$

Multiplying by $\kappa_\varepsilon > 0$ and using $\kappa_\varepsilon \le \alpha/4 \le 1/4$ gives

$$2 \le \kappa_\varepsilon K < 2 + \kappa_\varepsilon \le \frac{9}{4} \le 3.$$

Thus, we have $2 \le \kappa_\varepsilon K \le 3$.

Next, whenever $\lambda^2 \mathbb{E}[\|x_k\|_2^2] \le r^2$ holds, the choices of α, r, and κ_ε imply

$$\eta p_u \alpha \sigma^2 + \frac{2\eta\lambda^2}{p_l}\mathbb{E}[\|x_k\|_2^2] \le \frac{\eta p_l \mu^2 \varepsilon^2}{128} + \frac{2\eta r^2}{p_l} = \frac{\eta p_l \mu^2 \varepsilon^2}{64} = \frac{\kappa_\varepsilon \varepsilon}{4}. \tag{36}$$

We now prove the norm bound in (32) by contradiction. The bound trivially holds at index 0. Suppose that $k \le K$ is the first index at which it fails; then $k \ge 1$. For every $0 \le s < k$, the norm bound and the choice of λ give $\lambda^2 \mathbb{E}[\|x_s\|_2^2] \le r^2$. Note that since $d \ge 1$ and $\|x_0\|_2 \le \sqrt{d}\|x_0\|_\infty$, we have

$$\|x_0\|_2 + \sqrt{40\eta p_u \varepsilon/\kappa_\varepsilon}(1 + \lceil \log_2 K \rceil) \le \sqrt{d}(\|x_0\|_\infty + \sqrt{40\eta p_u \varepsilon/\kappa_\varepsilon}(1 + \lceil \log_2 K \rceil)).$$

Therefore, the choice of λ in (31) also satisfies (12). Thus (36) applies throughout this prefix. Dropping all decrease terms in (20) and using (36), yields

$$\mathbb{E}[\Phi_s] \le \mathbb{E}[\Phi_0] + \frac{s\kappa_\varepsilon\varepsilon}{4} \le \mathbb{E}[\Phi_0] + \frac{3\varepsilon}{4} < \mathbb{E}[\Phi_0] + 1, \quad 0 \le s \le k,$$

where we used $K\kappa_\varepsilon \le 3$ and $0 < \varepsilon \le 1$. This estimate includes index k, since it uses the remainder bound only at indices smaller than k.

Combining (20) with the remainder bound (36) and the KŁ estimate $\mathbb{E}[\|\nabla f(x_s)\|_2^2] \ge \mu^2 F_s^2$, we obtain

$$\mathbb{E}[\Phi_{s+1}] \le \mathbb{E}[\Phi_s] - \frac{\eta p_l \mu^2}{4} F_s^2 - \frac{\alpha}{2} V_s - \frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle] + \frac{\kappa_\varepsilon \varepsilon}{4}, \quad 0 \le s < k.$$

We use this inequality in two ways.

First, remove the term $-\frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle]$ from its right-hand side. It only increases the right-hand side because $P_s \succeq 0$ implies $\mathbb{E}[\langle m_s, P_s m_s\rangle] \ge 0$. For the remaining gradient and momentum-error terms, (27) gives

$$-\frac{\eta p_l \mu^2}{4} F_s^2 - \frac{\alpha}{2} V_s \le -\frac{\kappa_\varepsilon}{\varepsilon} \left(\mathbb{E}[\Phi_s]\right)^2.$$

The hypotheses of this quadratic bound hold because $\mathbb{E}[\Phi_s] < \mathbb{E}[\Phi_0] + 1$ on the prefix and we have already proved

$$\frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1) \le \frac{\alpha}{4}.$$

Thus, removing the update-energy term and then applying the quadratic bound yields the explicit bound

$$\begin{aligned}
\mathbb{E}[\Phi_{s+1}] &\le \mathbb{E}[\Phi_s] - \frac{\eta p_l \mu^2}{4} F_s^2 - \frac{\alpha}{2} V_s - \frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle] + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&\le \mathbb{E}[\Phi_s] - \frac{\eta p_l \mu^2}{4} F_s^2 - \frac{\alpha}{2} V_s + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&\le \mathbb{E}[\Phi_s] - \frac{\kappa_\varepsilon}{\varepsilon} \left(\mathbb{E}[\Phi_s]\right)^2 + \frac{\kappa_\varepsilon \varepsilon}{4}, \quad 0 \le s < k.
\end{aligned}$$

The last line is the scalar recursion (28). Thus Lemma 11, together with $\mathbb{E}[\Phi_s] \le \varepsilon/2 + Y_s$, yields

$$\mathbb{E}[\Phi_s] \le \frac{\varepsilon}{2} + \frac{\mathbb{E}[\Phi_0] + 1}{1 + \frac{\kappa_\varepsilon}{\varepsilon}(\mathbb{E}[\Phi_0] + 1)s}, \quad 0 \le s \le k.$$

In particular, this verifies (10) on the required prefix.

Second, start again from (20), with the KŁ estimate and the remainder bound (36). This time, remove the terms $-\frac{\eta p_l \mu^2}{4} F_s^2$ and $-\frac{\alpha}{2} V_s$ from the right-hand side, while leaving $-\frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle]$ in place. Both removed terms are nonpositive because $F_s^2 \ge 0$ and $V_s \ge 0$. Consequently,

$$\begin{aligned}
\mathbb{E}[\Phi_{s+1}] &\le \mathbb{E}[\Phi_s] - \frac{\eta p_l \mu^2}{4} F_s^2 - \frac{\alpha}{2} V_s - \frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle] + \frac{\kappa_\varepsilon \varepsilon}{4} \\
&\le \mathbb{E}[\Phi_s] - \frac{\eta}{8}\mathbb{E}[\langle m_s, P_s m_s\rangle] + \frac{\kappa_\varepsilon \varepsilon}{4}, \quad 0 \le s < k.
\end{aligned}$$

The last line verifies (9). Together with (10) and the previously verified parameter conditions, this establishes the hypotheses of Lemma 7. That lemma therefore proves the norm bound at index k, contradicting the assumption that the bound is not satisfied at iteration k as the first failing index. The norm bound consequently holds for all $0 \le k \le K$.

Thus the norm bound holds on the whole horizon, and the same estimates prove the bound for Y_k in (32). At $k = K$, we have

$$Y_K \le \frac{\varepsilon}{\kappa_\varepsilon K} \le \frac{\varepsilon}{2}.$$

As in the original argument,

$$\mathbb{E}[f(x_K) - f^\star] \le \mathbb{E}[\Phi_K] \le \frac{\varepsilon}{2} + Y_K \le \varepsilon.$$

This concludes the proof. □

E.3 Error Floor and BST Scaling Rule for AdamW

Corollary 4 (Full statement of Corollary 2)**.** *Under the setup of Theorem 5, assume that the token horizon T is large enough that*

$$T > \max\left\{\frac{128\sqrt{2}p_u^{3/2}(2L+\mu^2)B}{p_l^{3/2}\mu^2}, \frac{16384\sqrt{2}p_u^{5/2}(2L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4}\right\}, \tag{37}$$

and $\sigma^2 = \sigma_\star^2/B$ in Assumption 3. Set the hyperparameters of `AdamW` *(Algorithm 2) such that*

$$\alpha = \min\left\{1, \frac{p_l\mu^2 B\varepsilon^2}{128p_u\sigma_\star^2}\right\}, \quad \eta = \frac{\sqrt{p_l}\alpha}{4\sqrt{2}p_u^{3/2}(2L+\mu^2)},$$

$$1-\alpha \le 1-\beta \le \sqrt{1-\alpha}, \quad \lambda = \min\left\{\frac{1}{2\eta}, \frac{p_l\mu\varepsilon}{16\sqrt{d}\left[\|x_0\|_\infty + \sqrt{\frac{640p_u}{p_l\mu^2}}\left(1+\lceil\log_2(T/B)\rceil\right)\right]}\right\}.$$

Then, after T/B iterations, `AdamW` *achieves the optimization error*

$$\varepsilon = \max\left\{\frac{128\sqrt{2}p_u^{3/2}(2L+\mu^2)B}{p_l^{3/2}\mu^2 T}, \left(\frac{16384\sqrt{2}p_u^{5/2}(2L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4 T}\right)^{1/3}\right\}$$

Moreover, the BST scaling rules for `AdamW` *is given by*

$$\varepsilon \propto T^{-1/3}\left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4}\right)^{1/3}, \quad B \propto T^{2/3}\left(\frac{p_l\mu\sigma_\star}{p_u(L+\mu^2)}\right)^{2/3}, \quad \eta, \alpha \propto \textit{Const},$$

and

$$\lambda \propto \frac{1}{\sqrt{d}T^{1/3}}\frac{1}{(\|x_0\|_\infty + \sqrt{p_u/(p_l\mu^2)})}\left(\frac{p_l^{1/2}p_u^{5/2}(L+\mu^2)\sigma_\star^2}{\mu}\right)^{1/3},$$

up to logarithmic terms.

Proof. Theorem 5 uses

$$K = \left\lceil\frac{2}{\kappa_\varepsilon}\right\rceil, \quad \kappa_\varepsilon = \frac{\eta p_l\mu^2\varepsilon}{16}.$$

Consequently,

$$K = \left\lceil\frac{32}{\eta p_l\mu^2\varepsilon}\right\rceil.$$

The theorem also specifies the choice of α and η as follows

$$\alpha = \min\left\{1, \frac{p_l\mu^2\varepsilon^2}{128p_u\sigma^2}\right\}, \quad \eta = \frac{\sqrt{p_l}\alpha}{4\sqrt{2}p_u^{3/2}(2L+\mu^2)}.$$

Since $\alpha \le p_l\mu^2\varepsilon^2/(128p_u\sigma^2)$ and $0 < \varepsilon < 1$ (we show this later), we have

$$\frac{\sqrt{p_l}\alpha}{4\sqrt{2}p_u^{3/2}(2L+\mu^2)} \le \frac{2\alpha}{p_l(2L+\mu^2)} \le \frac{\mu^2\varepsilon^2}{64p_u\sigma^2(2L+\mu^2)} \le \frac{\varepsilon}{32p_u\sigma^2}.$$

Thus the variance-dependent learning-rate restriction is redundant, and we obtain a simplified expression for η

$$\eta = \frac{\sqrt{p_l}\alpha}{4\sqrt{2}p_u^{3/2}(2L+\mu^2)}.$$

Using

$$\frac{1}{\alpha} = \max\left\{1, \frac{128p_u\sigma^2}{p_l\mu^2\varepsilon^2}\right\},$$

we obtain the exact iteration formula

$$K = \max\left\{\frac{128\sqrt{2}p_u^{3/2}(2L+\mu^2)}{p_l^{3/2}\mu^2\varepsilon}, \frac{16384\sqrt{2}p_u^{5/2}(2L+\mu^2)\sigma^2}{p_l^{5/2}\mu^4\varepsilon^3}\right\}, \tag{38}$$

up to rounding errors. Multiplying by B and substituting $\sigma^2 = \sigma_\star^2/B$, the token budget T should satisfy,

$$T = \max\left\{\frac{128\sqrt{2}p_u^{3/2}(2L+\mu^2)B}{p_l^{3/2}\mu^2\varepsilon}, \frac{16384\sqrt{2}p_u^{5/2}(2L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4\varepsilon^3}\right\}.$$

For the fixed token budget T, the theorem guarantees the following target accuracy

$$\varepsilon = \max\left\{\frac{128\sqrt{2}p_u^{3/2}(2L+\mu^2)B}{p_l^{3/2}\mu^2T}, \left(\frac{16384\sqrt{2}p_u^{5/2}(2L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4T}\right)^{1/3}\right\}. \tag{39}$$

By the choice of T in (37), we obtain that $0 < \varepsilon < 1$ as required by Theorem 5.

We next derive the BST rules for hyperparameters. We want to choose the batch size as large as possible so that we stay in the batch-size-independent regime in (39), where

$$\varepsilon \cong \left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4T}\right)^{1/3}.$$

This means that we should choose B such that the two terms in (39) are equal, namely,

$$\frac{p_u^{3/2}(L+\mu^2)B}{p_l^{3/2}\mu^2T} \cong \left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4T}\right)^{1/3} \iff B \cong T^{2/3}\times\left(\frac{p_l\mu\sigma_\star}{p_u(L+\mu^2)}\right)^{2/3}.$$

where we ignore numerical constants. This choise of B implies that the momentum hyperparameter α is fixed under BST rule. Indeed, we have

$$\alpha \cong \frac{p_l\mu^2\varepsilon^2B}{p_u\sigma_\star^2} = \frac{p_l\mu^2}{p_u\sigma_\star^2}\times\left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4T}\right)^{2/3}\times T^{2/3}\left(\frac{p_l\mu\sigma_\star}{p_u(L+\mu^2)}\right)^{2/3} = \text{Const.}$$

This implies that the learning rate should remain constant as well, since

$$\eta \cong \frac{\sqrt{p_l}\alpha}{p_u^{3/2}(L+\mu^2)}.$$

Finally, the choice of λ in the statement of the corollary implies that

$$\begin{aligned}\lambda &\cong \frac{p_l\mu\varepsilon}{\sqrt{d}(\|x_0\|_\infty + \sqrt{p_u/(p_l\mu^2)})(1+\log_2(T/B)}\\ &= \frac{p_l\mu}{\sqrt{d}T^{1/3}}\frac{1}{(\|x_0\|_\infty + \sqrt{p_u/(p_l\mu^2)})(1+\log_2(T/B)}\times\left(\frac{p_u^{5/2}(L+\mu^2)\sigma_\star^2}{p_l^{5/2}\mu^4}\right)^{1/3}\\ &\cong \frac{1}{\sqrt{d}T^{1/3}}\frac{1}{(\|x_0\|_\infty + \sqrt{p_u/(p_l\mu^2)})(1+\log_2(T/B)}\left(\frac{p_l^{1/2}p_u^{5/2}(L+\mu^2)\sigma_\star^2}{\mu}\right)^{1/3}.\end{aligned}$$

Ignoring the logarithmic term, we arrive to

$$\lambda \cong \frac{1}{\sqrt{d}T^{1/3}}\frac{1}{(\|x_0\|_\infty + \sqrt{p_u/(p_l\mu^2)})}\left(\frac{p_l^{1/2}p_u^{5/2}(L+\mu^2)\sigma_\star^2}{\mu}\right)^{1/3}.$$

□

F μ-KŁ GUARANTEES FOR NEURAL NETWORKS

In this section we discuss the applicability of Assumption 2 on neural network training. We start by restating the assumption for arbitrary dual gradient norms:

Assumption. *The function $f : \mathcal{X} \to \mathbb{R}$ satisfies μ-KŁ in $\mathcal{X}$ if for some $\mu > 0$, for all $x \in \mathcal{X}$ we have*

$$\|\nabla f(x)\|_* \geq \mu(f(x) - f^\star).$$

Whenever we do not specify the norm, we use the Euclidean norm, which is self-dual and which is the norm in which our `SGDM` and `AdamW` guarantees are stated. For a dual norm with $\|x\|_* \geq c_* \|x\|_2$, all bounds below hold with μ replaced by $c_*\mu$. In Section F.3, we discuss that a geometry aware formulation of μ-KŁ can be useful for invariance of μ under reparameterizations.

As the optimization process converges to any minimizer, including the global minimizer $x^\star$, there is a result from Bochnak & Łojasiewicz (1971) that proves that there exists a region that satisfies the μ-KŁ condition around local minima. The formal statement is the following:

Lemma 12 (Bochnak–Łojasiewicz radial inequality)**.** *Let g be real analytic near $0 \in \mathbb{R}^d$, with $g(0) = 0$. If g is not locally constant and its first nonzero homogeneous Taylor term has degree ν, then, for every fixed $c \in (0, \nu)$, there is $r_c > 0$ such that*

$$\|u\|_2 \, \|\nabla g(u)\|_2 \geq c|g(u)| \qquad (\|u\|_2 < r_c). \tag{40}$$

If g is locally zero, the inequality is trivial.

This original inequality covers all real analytic functions and proves that, in some neighborhood around the optimum, the μ-KŁ inequality is satisfied. This result was later extended in Loi (2016) to include local radial bounds for C^1 functions, encompassing sub-analytic functions. So for a global minimum x^*, with value $f(x^*) = f^\star$, one can set $g(x) = f(x + x^*) - f(x^*)$ and substitute in Lemma 12 to get μ-KŁ for $\mu = c/r_c$.

However, this general bound fails to quantify the size of the benign neighborhood r_c around the optimum, as well as the constants in μ precisely. Additionally it is important to show that once the algorithm enters this region, the algorithm stays in that region in later iterations. In the following sections we give explicit constants on specific regions and in Section F.1 we also show that the condition is preserved after entry.

Uniform μ-KŁ guarantee for two layer linear nets. Specifically in Section F.1, we obtain μ in closed form for two-layer linear networks trained by gradient flow under MSE loss. The analysis captures an initial saddle close to which μ degrades with decreasing initialization scale. Once the flow enters the low-loss region, the guarantee persists and we can formulate a μ-KŁ inequality over the remaining training trajectory. We also demonstrate how even in this example the KŁ inequality allows us to model a more general setting in comparison to the PŁ one.

μ-KŁ guarantee under readout separability. In Section F.2, we turn to the readout geometry of a late training stage. Assuming a bounded readout and bounded features, as commonly satisfied when using a normalization layer before the readout and training with Scion or Adam with weight decay. We start with a conditional guarantee relative to the best readout for the current features, and end with proper μ-KŁ guarantees, when assuming a margin or a separator given the features, capturing a stage where the features are rich enough to model the data and the readout still needs to be learned. Indeed, Tinaz et al. (2026) show that in the initial phase of training hidden weights progressively align with the signal directions before the readout layer grows towards the optimum.

Geometry-aware μ-KŁ. In Section F.3, we provide a μ-KŁ bound for 2-layer networks, where gradient flow converges to Wasserstein gradient flow in the mean-field limit. The bound does not improve over those provided in Section F.2, but the analysis illustrates that using a geometry-aware variant of the μ-KŁ condition can make the μ constant invariant to reparameterizations.

μ-KŁ guarantee in the NTK regime. Finally, in Section F.4, we cover a complementary setting, namely overparameterized networks in the NTK regime trained with cross-entropy (CE) loss sufficiently close to initialization. There the assumption reduces to a lower bound on the minimal NTK eigenvalue, required to hold with high probability over the initialization. Corollary 7 reduces the assumption further to a spectral condition on the data alone.

μ-KŁ is weaker and more practical than PŁ. If $\|\nabla f(x)\|_2^2 \geq 2\beta\big(f(x) - f^\star\big)$ on $\mathcal{X}$ and near optimality $f - f^\star \leq D$ on $\mathcal{X}$, then Assumption 2 holds on $\mathcal{X}$ with $\mu = \sqrt{2\beta/D}$, since $\sqrt{f - f^\star} \geq {(f-f^\star)}/{\sqrt{D}}$. Thus, for fixed constants, Assumption 2 is weaker than Polyak–Łojasiewicz (Polyak, 1963; Karimi et al., 2016) near the optimum. Example 7 shows that the PŁ condition does not apply in neural networks for which μ-KŁ applies. This makes the μ-KŁ assumption a more applicable condition for studying neural network training dynamics.

Before proceeding we introduce some additional notation needed for this section.

Notation. Let $\{(s_i, y_i)\}_{i=1}^n$, $s_i \in \mathbb{R}^{d_x}$ and $y_i \in [C] \coloneqq \{1, \dots, C\}$, be a labeled training set, let $z_i(x) \in \mathbb{R}^C$ be the logits at a data point (s_i, y_i), and let $p_i(x) \in \mathbb{R}^C$ be the softmax vector with entries $p_{i,c}(x) = {e^{z_{i,c}(x)}}/{\sum_{c' \in [C]} e^{z_{i,c'}(x)}}$, so that p_{i,y_i} is the probability assigned to the correct class. We stack the logits as $z(x) \coloneqq \big(z_1(x)^\top, \dots, z_n(x)^\top\big)^\top \in \mathbb{R}^{nC}$, and define the loss as a function of parameters $x \in \mathbb{R}^d$ and of logits z as

$$f(x) = F(z(x)) = \frac{1}{n}\sum_{i=1}^n \ell(z_i(x), y_i), \qquad \ell(z_i, y_i) = -\log \frac{e^{z_{i,y_i}}}{\sum_{c \in [C]} e^{z_{i,c}}},$$

abbreviating $\ell_i \coloneqq \ell(z_i(x), y_i)$. Here F always denotes the loss as a function of logits and f the loss as a function of parameters, and $f^\star \coloneqq \inf_{x \in \mathbb{R}^d} f(x)$. Define the n-fold product of the zero-sum hyperplane

$$\mathcal{S} \coloneqq \big\{u \in (\mathbb{R}^C)^n \mid \sum_{c \in [C]} u_{i,c} = 0 \text{ for every } i \in [n]\big\}.$$

Since $\nabla_z \ell(z_i(x), y_i) = p_i - e_{y_i}$, the loss-logit gradients lie in $\mathcal{S}$, i.e., $\nabla_z F(z(x)) \in \mathcal{S}$. We write $\mathcal{X}_\varepsilon \coloneqq \mathcal{X} \cap \{x \mid f(x) - f^\star \geq \varepsilon\}$ for the part of a region $\mathcal{X}$ with suboptimality at least ε. Within this appendix, m is a network width and σ an activation, while $\sigma_j(\cdot)$ and $\sigma_{\min}(\cdot)$ are singular values.

F.1 Two-Layer Linear Networks

We begin with the setting in which μ is available in closed form, namely a two-layer linear network trained with gradient flow under squared loss. We follow an approach similar to Min et al. (2021), followed by a lower bound on those singular values in terms of the layer imbalance conserved by gradient flow (Arora et al., 2018; 2019).

Setup. Let $S \in \mathbb{R}^{n \times d_x}$ stack the inputs s_i and $Y \in \mathbb{R}^{n \times d_x}$ the targets, and assume whitened inputs, $\frac{1}{n} S^\top S = I$. With parameters $x = (W_1, W_2) \in \mathbb{R}^{d_x \times m} \times \mathbb{R}^{m \times d_x}$, width $m \geq d_x$, and end-to-end matrix $\Theta(x) \coloneqq W_1 W_2$,

$$\begin{aligned} \Theta^\star = \frac{1}{n} S^\top Y, \quad E(x) = \Theta^\star - \Theta(x), \\ f(x) = \frac{1}{2n}\|Y - S W_1 W_2\|_F^2, \qquad \Delta(x) \coloneqq f(x) - f^\star = \frac{1}{2}\|E(x)\|_F^2. \end{aligned} \tag{41}$$

where $E(x)$ is the residual and $\Theta^\star$ the global minimizer. Whitening requires $n \geq d_x$, and Lemma 13 requires $m \geq d_x$. Input and target dimensions are equal so that both factors can be full rank.

Lemma 13 (Gradient identity and spectral lower bound)**.** *Let $\sigma_j(E(x))$ denote the singular values of $E(x)$, with singular vectors u_j, v_j. Then,*

$$\begin{aligned} \|\nabla f(x)\|_2^2 &= \sum_{j \in [d_x]} \sigma_j(E(x))^2 \left(\big\|W_1^\top u_j\big\|_2^2 + \|W_2 v_j\|_2^2\right) \\ &\geq 2\left(\sigma_{\min}(W_1)^2 + \sigma_{\min}(W_2)^2\right)(f(x) - f^\star). \end{aligned} \tag{42}$$

Proof. From differentiation we have $\nabla_{W_1} f(x) = -E(x) W_2^\top$ and $\nabla_{W_2} f(x) = -W_1^\top E(x)$, and

$$\begin{aligned} \|\nabla f(x)\|_2^2 &= \big\|E(x) W_2^\top\big\|_F^2 + \big\|W_1^\top E(x)\big\|_F^2 \\ &= \mathrm{Tr}(E^\top E\, W_2^\top W_2) + \mathrm{Tr}(E E^\top W_1 W_1^\top). \end{aligned}$$

Substituting $E^\top E = \sum_j \sigma_j^2 v_j v_j^\top$ and $EE^\top = \sum_j \sigma_j^2 u_j u_j^\top$ gives the identity. As $m \geq d_x$, every unit vector satisfies $\|W_1^\top u\|_2 \geq \sigma_{\min}(W_1)$ and $\|W_2 v\|_2 \geq \sigma_{\min}(W_2)$. The inequality follows from $\sum_j \sigma_j^2 = \|E(x)\|_F^2 = 2(f(x) - f^\star)$. □

Lemma 14 (The imbalance controls the factors)**.** *Define the imbalance:*

$$\Xi(x) \coloneqq W_1^\top W_1 - W_2 W_2^\top \in \mathbb{R}^{m \times m}. \tag{43}$$

For every x, we have:

$$\sigma_{\min}(W_1)^2 + \sigma_{\min}(W_2)^2 \geq \sqrt{\|\Xi(x)\|_{\mathrm{op}}^2 + 4\,\sigma_{\min}(\Theta(x))^2} - \|\Xi(x)\|_{\mathrm{op}}. \tag{44}$$

Proof. Write $\Xi = \Xi(x)$ and $t = \sigma_{\min}(\Theta(x))$. Let v be a unit right singular vector of W_2 for $\nu_2 \coloneqq \sigma_{\min}(W_2)$, so $W_2^\top W_2 v = \nu_2^2 v$. Denoting $w \coloneqq W_2 v$ gives $\|w\|_2 = \nu_2$ and $\|W_2^\top w\|_2 = \nu_2^2$. Using $W_1^\top W_1 = W_2 W_2^\top + \Xi$,

$$t^2 \leq \|\Theta(x) v\|_2^2 = \|W_1 w\|_2^2 = \|W_2^\top w\|_2^2 + w^\top \Xi w \leq \nu_2^4 + \|\Xi\|_{\mathrm{op}} \nu_2^2.$$

Solving $\nu_2^4 + \|\Xi\|_{\mathrm{op}} \nu_2^2 - t^2 \geq 0$ for ν_2^2 gives $\sigma_{\min}(W_2)^2 \geq \frac{1}{2}\big(\sqrt{\|\Xi\|_{\mathrm{op}}^2 + 4t^2} - \|\Xi\|_{\mathrm{op}}\big)$. Symmetrically, let u be a unit left singular vector of W_1 for $\nu_1 \coloneqq \sigma_{\min}(W_1)$ and denote $\zeta \coloneqq W_1^\top u$, so $\|\zeta\|_2 = \nu_1$ and $\|W_1 \zeta\|_2 = \nu_1^2$. Using $W_2 W_2^\top = W_1^\top W_1 - \Xi$,

$$t^2 \leq \|\Theta(x)^\top u\|_2^2 = \|W_2^\top \zeta\|_2^2 = \|W_1 \zeta\|_2^2 - \zeta^\top \Xi \zeta \leq \nu_1^4 + \|\Xi\|_{\mathrm{op}} \nu_1^2,$$

giving the same bound for $\sigma_{\min}(W_1)^2$. Adding the two proves (44). □

Lemma 14 is an operator-norm consequence of the factor estimate in (Min et al., 2021, Lemma 8), specialized to the single quantity $\|\Xi\|_{\mathrm{op}}$ in place of the two extreme eigenvalues of Ξ.

Lemma 15 (Lower bound of smallest singular value)**.** *For all x, we have that*

$$\sigma_{\min}(\Theta(x)) \geq \sigma_{\min}(\Theta^\star) - \sqrt{2\Delta(x)}. \tag{45}$$

In particular, if $0 < D < \sigma_{\min}(\Theta^\star)^2/2$ and $\Delta(x) \leq D$, then $\sigma_{\min}(\Theta(x)) \geq \theta_D \coloneqq \sigma_{\min}(\Theta^\star) - \sqrt{2D} > 0$.

Proof. For every unit vector v, the reverse triangle inequality gives

$$\|\Theta(x) v\|_2 = \|\Theta^\star v - E v\|_2 \geq \sigma_{\min}(\Theta^\star) - \|E\|_{\mathrm{op}} \geq \sigma_{\min}(\Theta^\star) - \|E\|_F.$$

Minimize over v and use (41), this allows us to get the statement. □

Theorem 6 (μ-KŁ on a low-loss set)**.** *For a constant $M_\Xi \in [0, \infty)$ a level $D \in \big(0, \frac{1}{2}\sigma_{\min}(\Theta^\star)^2\big)$, and let $\theta_D \coloneqq \sigma_{\min}(\Theta^\star) - \sqrt{2D} > 0$. Then, μ-KŁ holds on*

$$\mathcal{X} \coloneqq \big\{x \mid \Delta(x)\|\Xi(x)\|_{\mathrm{op}} \leq M_\Xi,\ \Delta(x) \leq D\big\} \quad \textit{with} \quad \mu_{D, M_\Xi} = \left(\frac{8\theta_D^2}{\sqrt{M_\Xi^2 + 4\theta_D^2 D^2} + M_\Xi}\right)^{1/2}$$

Proof. For $\Delta(x) = 0$, the inequalities hold. So we assume $\Delta(x) > 0$ for the rest of the proof. Let $c = \|\Xi(x)\|_{\mathrm{op}}$. Lemmas 13 to 15 give

$$\|\nabla f(x)\|_2^2 \geq 2\Delta(x)\big(\sqrt{c^2 + 4\theta_D^2} - c\big).$$

Multiplying by $\frac{\sqrt{c^2+4\theta_D^2}+c}{\Delta^2}$ yields

$$\begin{aligned}
\frac{\|\nabla f(x)\|_2^2}{\Delta^2} &\geq \frac{8\theta_D^2}{\Delta(\sqrt{c^2 + 4\theta_D^2} + c)} \\
&= \frac{8\theta_D^2}{\sqrt{(c\Delta)^2 + 4\theta_D^2\Delta^2} + c\Delta} \\
&\geq \frac{8\theta_D^2}{\sqrt{M_\Xi^2 + 4\theta_D^2 D^2} + M_\Xi}.
\end{aligned} \tag{46}$$

The final denominator is positive because $D, \theta_D > 0$, even if $M_\Xi = 0$. Taking square roots proves the theorem. □

The theorem is pointwise and does not assert that every gradient-flow trajectory reaches the level D: for example, the choice $W_1 = W_2 = 0$ is a stationary saddle whenever $\Theta^\star \neq 0$. The admissible threshold therefore only depends on $\frac{1}{2}\sigma_{\min}(\Theta^\star)^2$.

Corollary 5 (Along any flow that enters the low-loss region)**.** *Let x_t solve $\dot{x}_t = -\nabla f(x_t)$ from an arbitrary x_0. If $f(x_{t_0}) - f^\star \leq D$ for some $t_0 \geq 0$ and some $D \in \left(0, \frac{1}{2}\sigma_{\min}(\Theta^\star)^2\right)$, then μ-KŁ holds at x_t for every $t \geq t_0$, with the constant $\mu = \mu_{D,M_\Xi}$ of Theorem 6.*

Proof. Along the flow $\frac{d}{dt}(W_1^\top W_1) = W_2 E^\top W_1 + W_1^\top E W_2^\top = \frac{d}{dt}(W_2 W_2^\top)$, so $\Xi(x_t) = \Xi(x_0)$ for all t. As $f(x_t)$ is nonincreasing, $x_t \in \mathcal{X}$ for every $t \geq t_0$,since:

$$\Delta(x_t)\|\Xi(x_t)\|_{\mathrm{op}} = \Delta(x_t)\|\Xi(x_0)\|_{\mathrm{op}} \leq \Delta(x_0)\|\Xi(x_0)\|_{\mathrm{op}} \leq M_\Xi$$

□

The only property of the initialization that enters is $\|\Xi(x_0)\|_{\mathrm{op}}$, which is directly measurable and conserved throughout training. We impose no alignment, balance or scale condition. The constant in Theorem 6 has two regimes, depending on $\|\Xi\|_{\mathrm{op}}$ relative to θ_D. In the balanced regime $\|\Xi\|_{\mathrm{op}} \ll \theta_D$ the square root is dominated by θ_D and $\mu \approx 2\sqrt{\theta_D/D}$, which is the best case attained by exact balance. In the imbalanced regime $\|\Xi\|_{\mathrm{op}} \gg \theta_D$, the two terms nearly cancel and $\mu \approx 2\theta_D/\sqrt{D\,\|\Xi\|_{\mathrm{op}}}$, degrading as $\|\Xi\|_{\mathrm{op}}^{-1/2}$. At the usual $1/\sqrt{\text{fan-in}}$ scale $\|\Xi(x_0)\|_{\mathrm{op}}$ is of order one or larger, so the imbalanced regime is the relevant one in practice.

A scale-uniform lower bound on μ must exclude the small-scale neighborhood of the zero-factor saddle. If both factors are initialized at scale ς, then $\|\nabla f(x_0)\|_2 = \mathcal{O}(\varsigma)$ while $f(x_0) - f^\star \to \frac{1}{2}\|\Theta^\star\|_F^2$, so no constant independent of ς can hold at x_0. In this setting, μ degrades linearly with the initialization scale. If the loss drops below $\frac{1}{2}\sigma_{\min}(\Theta^\star)^2$, Theorem 6 guarantees μ-KŁ for the rest of the trajectory. The constant improves as the chosen loss level D falls, and with smaller conserved initialization imbalance. For an aligned, balanced initialization the flow is solvable in closed form (Fukumizu, 1998; Saxe et al., 2014; Gidel et al., 2019; Bach, 2026) and exact solutions and dynamics are available for unbalanced initializations as well (Tarmoun et al., 2021; Braun et al., 2022; Kunin et al., 2024).

Recall that KŁ is a weaker assumption than PŁ near the optimum. Theorem 6 establishes μ-KŁ on a low-loss set. The following example shows that such a set need not admit a positive uniform PŁ constant, showing that KŁ is indeed strictly more applicable in neural network analyses.

Example 7 (KŁ applies, but PŁ fails to hold in 2-layer network)**.** *Take $\Theta^\star = I_2$ and, for $0 < t \leq 1$, set*

$$W_1(t) = \begin{pmatrix} t & 0 \\ 0 & t^{-1} \end{pmatrix}, \qquad W_2(t) = \begin{pmatrix} [t(1+t^2)]^{-1} & -(1+t^2)^{-1} \\ 0 & t \end{pmatrix}.$$

Direct multiplication and the gradient formulas give

$$\Delta(t) = \frac{t^2}{2(1+t^2)}, \qquad \|\nabla f(x_t)\|_2^2 = \frac{t^4(2+t^2)}{(1+t^2)^2}.$$

In particular,

$$\frac{\|\nabla f(x_t)\|_2}{\Delta(t)} = 2\sqrt{2+t^2} \geq 2\sqrt{2}, \qquad \frac{\|\nabla f(x_t)\|_2^2}{2\Delta(t)} = \frac{t^2(2+t^2)}{1+t^2} \longrightarrow 0.$$

This example satisfies the weighted imbalance hypothesis of our statements. Since $\Delta(t) \leq 1/4$ and $\Delta(t)\|\Xi(x_t)\|_{\mathrm{op}} \leq 1$, Theorem 6 applies, and the function is KŁ but not PŁ, under the theorem's assumptions. The factors diverge as $t \to 0$. A more complicated example of KŁ satisfied with PŁ failing is given in our lower bound section.

F.2 Analysis for a Linear Readout

We now consider a late training stage and formulate results that can also be applied to practical architectures, in a stage where the last-layer representations are sufficiently rich. In such a scenario we can assume an independently trainable linear readout, such cases include the untied-head Peri-LN experiments. So let:

$$x = (A, w), \qquad A \in \mathbb{R}^{h\times C}, \qquad z_i(A, w) = A^\top \psi_i(w), \qquad \psi_i(w) \in \mathbb{R}^h, \tag{47}$$

so that the parameters are $x = (A, w)$ with A the linear readout and w the feature parameters. We write $f(x) = f(A, w)$.

We assume bounded features, $\|\psi_i(w)\|_2 \le \varrho$, and a bounded readout, $\|A\|_F \le \tau_A$, on the region under consideration. This is motivated by practical architectures, which typically use a RMSNorm layer before the readout, and readout weights are bounded under Muon/Scion as well as Adam with weight decay. The logit and loss envelopes depend only on the product $\kappa := \tau_A\varrho$, since rescaling $(A, \psi) \mapsto (cA, \psi/c)$ leaves every logit unchanged while sending $\tau_A \mapsto c\tau_A$ and $\varrho \mapsto \varrho/c$.

We formulate three complementary bounds. The first assumes only bounded features and readout. It controls the gap to the best loss reachable with the current features, so it certifies μ-KŁ for the global $f^\star$ only when those features can already express an optimal predictor, and otherwise only up to the amount by which they cannot. The second assumes in addition that the current logits separate the training data with a uniform margin $\gamma > 0$, and then guarantees the original μ-KŁ condition with respect to f^* on all of $\mathcal{X}$. The third weakens that margin hypothesis, asking only that the features admit a separator rather than that the current readout achieves one, but additionally imposes a restriction to a loss floor $\mathcal{X}_\varepsilon$.

Lemma 16 (Conditional readout bound)**.** *For every w and every A with $\|A\|_F \le \tau_A$, define $f^\star_w := \min_{\|A'\|_F\le\tau_A} f(A', w)$. Then*

$$f(A, w) - f^\star_w \;\le\; 2\tau_A \,\|\nabla_A f(A, w)\|_F \,. \tag{48}$$

Moreover, if $\|\psi_i(w)\|_2 \le \varrho$, then $\|z_i\|_\infty \le \kappa$ and $\ell_i \le \log(1 + (C - 1)e^{\sqrt{2}\kappa}) := M^\star$ for every i, so the loss bound M of Lemma 19 may be taken to be $M^\star$.

Proof. The minimization region for $\|A\|_F \le \tau_A$ is compact and the function is continuous, so there exists a minimizer $A^\star_w := \arg\min_{\|A'\|_F\le\tau_A} f(A', w)$. For fixed w the map $A \mapsto f(A, w)$ is cross-entropy composed with a linear map, hence convex and differentiable on all of $\mathbb{R}^{h\times C}$. So:

$$f(A, w) - f(A^\star_w, w) \le \langle \nabla_A f, A - A^\star_w\rangle$$

taking Cauchy–Schwarz and utilizing that:

$$\|A - A^\star_w\|_F \le \|A\|_F + \|A^\star_w\|_F \le 2\tau_A$$

gives (48). For the second claim, $|z_{i,c}| \le \|a_c\|_2 \,\|\psi_i\|_2 \le \kappa$, and cross-entropy depends on the logits only through the differences $z_{i,y_i} - z_{i,c} = (a_{y_i} - a_c)^\top \psi_i$, whose absolute values are at most $\sqrt{2}\kappa$. □

Inequality (48) is the Frank–Wolfe gap bound of Jaggi (2013, Section 2) specialized to a Frobenius ball. We state it to fix the constant, which is tight. The bound of Jaggi (2013) holds for an arbitrary compact convex set, so (48) remains valid for any norm ball once $\|\nabla_A f\|_F$ is replaced by the norm dual to the constraint. For `Scion` and related LMO methods the native ball is spectral, which gives the same constant with the nuclear norm on the gradient, and converting to the Euclidean norm of μ-KŁ costs at most $\sqrt{\min\{n, h, C - 1\}}$, the rank of $\nabla_A f$ being limited by the zero-sum structure of the loss-logit gradients. For the decoupled update $A_{t+1} = (1 - \eta\lambda)A_t - \eta u_t$, if $0 \le \eta\lambda \le 1$, $\lambda > 0$ and $\|u_t\|_F \le U_A$ uniformly, then

$$\|A_t\|_F \le \max\left\{\|A_0\|_F \,, \frac{U_A}{\lambda}\right\} \qquad \text{for all } t.$$

Thus `AdamW` yields a readout ball only together with a uniform bound on its preconditioned update. For example, a coordinate-wise bound of one gives $U_A \le \sqrt{hC}$. In either case, (48) gives a

gradient-domination constant $1/(2\tau_A)$ relative to $f_w^\star$, hence only a μ-KŁ guarantee if the optimum is reachable within the contraint set, $f_w^\star = f^\star$.

The next lemma formulates a μ-KŁ guarantee for the global $f^\star$, on all of $\mathcal{X}$, with no comparator and no sublevel-set exclusion, but assuming a margin.

Lemma 17 (Positive margin implies μ-KŁ)**.** *Suppose $\|A\|_F \le \tau_A$ on $\mathcal{X}$ and there is a uniform margin lower bound $\gamma > 0$ such that, for every $(A, w) \in \mathcal{X}$,*

$$z_{i,y_i}(A, w) - z_{i,c}(A, w) \ge \gamma, \qquad i \in [n], \quad c \ne y_i. \tag{49}$$

With $M_\gamma \coloneqq \log\big(1 + (C-1)e^{-\gamma}\big)$ and $q_{M_\gamma} = {}^{(1-e^{-M_\gamma})}/_{M_\gamma}$, μ-KŁ holds on all of $\mathcal{X}$ with

$$\mu = \frac{\gamma q_{M_\gamma}}{\tau_A}. \tag{50}$$

Proof. For fixed features (we omit w for compactness), we have that:

$$\nabla_A f = \frac{1}{n}\sum_{j=1}^{n} \psi_i \left(p_i - e_{y_i}\right)^T$$

Linearity in A gives

$$\begin{aligned} -\langle \nabla_A f, A\rangle &= \frac{1}{n}\sum_{i=1}^{n}\left(z_{i,y_i} - \sum_c p_{i,c} z_{i,c}\right) \\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{c\ne y_i} p_{i,c}\big(z_{i,y_i} - z_{i,c}\big) \\ &\ge \frac{\gamma}{n}\sum_{i\in[n]}(1 - p_{i,y_i}). \end{aligned}$$

Where in the second equality we used the fact that: $\sum_c p_{i,c} = 1$. Moreover, (49) implies $\ell_i \le M_\gamma$, and since $t \mapsto {}^{(1-e^{-t})}/_t$ is decreasing, $1 - p_{i,y_i} = 1 - e^{-\ell_i} \ge q_{M_\gamma}\ell_i$. Thus, using $\|A\|_F \le \tau_A$ and that $\nabla_A f$ is a block of $\nabla_x f$,

$$\tau_A \|\nabla_x f\|_2 \ge -\langle \nabla_A f, A\rangle \ge \gamma q_{M_\gamma} f \ge \gamma q_{M_\gamma}(f - f^\star). \qquad \square$$

Rearranging gives the lemma statement.

The margin hypothesis describes a late-stage region in which the current logits already separate the training data. It yields the most direct practical guarantee in this appendix: it allows arbitrary depth and feature learning, needs no NTK approximation, and can be checked from the training margins and readout norm.

Lemma 17 needs the current logits to be separated. To weaken this to the existence of a separator for the features, we use the following comparator bound, a quantitative form of weak quasi-convexity (Hardt et al., 2018). A radius bound together with a directional gap bound toward a near-optimal comparator yield μ-KŁ. Unlike Lemma 17, this requires staying sufficiently far from the optimum.

Lemma 18 (Comparator bound)**.** *Let $R > 0$ and $\Delta \ge 0$. Suppose that for every $x \in \mathcal{X}$ there is $\bar{x} = \bar{x}(x)$ with $f(\bar{x}) - f^\star \le \Delta$, $\|x - \bar{x}\|_2 \le R$, and*

$$f(x) - f(\bar{x}) \le \langle \nabla f(x), x - \bar{x}\rangle. \tag{51}$$

Then, for every $\varepsilon > \Delta$, μ-KŁ holds on $\mathcal{X}_\varepsilon = \{x \in \mathcal{X} : f(x) - f^\star \ge \epsilon\}$ with

$$\mu = \frac{1}{R}\Big(1 - \frac{\Delta}{\varepsilon}\Big). \tag{52}$$

Proof. By (51), we have:

$$\begin{aligned} f(x) - f^\star &\le \langle \nabla f(x), x - \bar{x}\rangle + f(\bar{x}) - f^\star \\ &\le \|\nabla f(x)\|\|x - \bar{x}\| + \Delta \\ &\le \|\nabla f(x)\| R + \Delta \\ &\le \|\nabla f(x)\| R + \frac{\Delta}{\epsilon}(f(x) - f^\star) \end{aligned}$$

In the second line we used Cauchy–Schwarz and the assumption $f(\bar{x}) - f^\star \le \Delta$. In the third line we used $\|x - \bar{x}\| \le R$ and in the last line $f(x) - f^\star \ge \varepsilon$. By rearranging the last inequality, we get the lemma statement. □

Note that for Lemma 18 we do not require global convexity, only that the directional gap condition (51) holds toward the comparator.

Lemma 17 requires the *current* logits to be separated, which is a hypothesis on the readout as much as on the features. The following corollary weakens it to a hypothesis on the features alone, namely that some readout of bounded norm would separate them. The resulting μ degrades only logarithmically as that restriction is pushed towards the optimum.

Corollary 6 (Separable features: μ^{-1} grows only logarithmically in $1/\varepsilon$)**.** *Suppose that $A \mapsto f(A, w)$ is convex for every fixed w, and that $\|A\|_F \le \tau_A$ for every $(A, w) \in \mathcal{X}$. Suppose further that, for every feature parameter w occuring in $\mathcal{X}$, there exists $\Omega(w) = [\omega_1, \dots, \omega_C] \in \mathbb{R}^{h \times C}$ such that*

$$\|\Omega\|_F \le R_\Omega, \qquad (\omega_{y_i} - \omega_c)^\top \psi_i(w) \ge \gamma > 0, \qquad i \in [n], c \ne y_i \tag{53}$$

Fix $a \in (0, 1)$, $\varepsilon > 0$ and $s > 0$ such that $\Delta := \log\big(1 + (C-1)e^{-s\gamma}\big) \le (1-a)\varepsilon$. Then μ-KŁ holds on $\mathcal{X}_\varepsilon$ with

$$\mu = \frac{a}{\tau_A + sR_\Omega}. \tag{54}$$

Proof. Take $\bar{x} = (s\Omega, w)$. Its margins are $s(\omega_{y_i} - \omega_c)^\top \psi_i(w) \ge s\gamma$, by assumption, so:

$$f(\bar{x}) - f^\star \le f(\bar{x}) \le \log(1 + (C-1)e^{-s\gamma}) \le (1-a)\varepsilon$$

also $\|A - s\Omega\|_F \le \tau_A + sR_\Omega := R$. Convexity in A provides the last necessary condition to apply Lemma 18, which we can apply with $\Delta = (1-a)\varepsilon$, $\varepsilon = \varepsilon$ and $R = \tau_A + sR_\Omega$, which gives $\mu = (1 - \Delta/\varepsilon)/R \ge a/R$ completing the proof. □

Let ε sufficiently small, then one may take

$$s \ge \frac{1}{\gamma} \log\Big(\frac{C-1}{e^{(1-a)\varepsilon} - 1}\Big)$$

whenever the right-hand side is positive, we have:

$$\mu^{-1} = \frac{\tau_A}{a} + \frac{R_\Omega}{a\gamma}\left[\log\frac{1}{\varepsilon} + \log\frac{C-1}{1-a} + \mathcal{O}(\varepsilon)\right]$$

which implies that: $\mu^{-1} = \mathcal{O}\big(\gamma^{-1} R_\Omega \log(1/\varepsilon)\big)$ as $\varepsilon \downarrow 0$. The hypothesis holds when the learned features separate the training data, as an interpolating overparameterized network does late in training on sufficiently learnable data distributions. Then γ and R_Ω are the margin and norm of any separator, both measurable after training. We need to assume that the readout is not yet sufficiently well optimized, so that we still incur a loss gap, $\mathcal{X}_\varepsilon$. The guarantee holds uniformly on $\mathcal{X}_\varepsilon$, where the loss is at least ε.

The bound (48), namely

$$f(A, w) - f_w^\star \le 2\tau_A \|\nabla_A f(A, w)\|_F$$

measures suboptimality against $f_w^\star$ rather than the global $f^\star$. Writing $\Delta_{\mathrm{feat}}(w) := f_w^\star - f^\star$ and $\delta := \sup_{(A,w)\in\mathcal{X}} \Delta_{\mathrm{feat}}(w)$, Lemma 18 applies with comparator $\bar{x} = (A_w^\star, w)$, radius $R = 2\tau_A$ and $\Delta = \delta$, and gives μ-KŁ on $\mathcal{X}_\varepsilon$ with $\mu = 1/(2\tau_A)(1 - \delta/\varepsilon)$ for every $\varepsilon > \delta$. The added hypothesis is that the features are uniformly close to optimal on $\mathcal{X}$. It is another feature-learning assumption that can be imposed. Corollary 6 provides a separate guarantee based on feature separability, but it does not imply that δ is small.

Bounded features and a bounded readout alone cannot imply μ-KŁ. We showcase this through the following example:

Example 8 (Bounded heads and features do not exclude bad stationary points)**.** *Suppose every empirical class frequency $\widehat{\pi}_c$ is positive and the features use pure radial normalization without learned affine parameters:*

$$\psi_i(w) = \varrho \frac{u_i(w)}{\|u_i(w)\|_2}, \qquad u_i(w) \neq 0.$$

Assume $\psi_i(w_0) = \psi$ for all i, where $\|\psi\|_2 = \varrho > 0$. Let

$$v = \log \widehat{\pi} - \frac{1}{C}\left(\sum_c \log \widehat{\pi}_c\right) \mathbf{1}_C, \qquad A_0 = \varrho^{-2}\psi v^\top.$$

If $\|v\|_2 / \varrho \leq \tau_A$, this point has a bounded head. Every logit vector is v, every prediction is $\widehat{\pi}$, and $f(A_0, w_0) = -\sum_c \widehat{\pi}_c \log \widehat{\pi}_c$. The head gradient is zero because $n^{-1}\sum_i(\widehat{\pi} - e_{y_i}) = 0$. For the feature block,

$$D_{u_i}\psi_i = \frac{\varrho}{\|u_i\|_2}(I - \widehat{u}_i\widehat{u}_i^\top), \qquad \nabla_{\psi_i} f = \frac{1}{n} A_0(\widehat{\pi} - e_{y_i}),$$

and the latter vector is parallel to ψ, hence to u_i. The projector annihilates it, so the full gradient is zero. Whenever the same model admits a predictor with smaller loss, this is a nonoptimal stationary point and no positive global-gap constant can hold on any region containing it. This is an exact conditional construction, not an assertion that such collapse occurs in a particular run or in stabilized normalization with learned affine parameters.

F.3 A Two-Layer Mean-Field Perspective

For a two-layer network, collect the parameters of one neuron in u and its logits on the training set in $\phi(u) \in \mathbb{R}^{nC}$. For a probability measure ρ over neurons, write

$$z(\rho) = \int \phi(u)\, d\rho(u), \qquad \mathcal{F}(\rho) = F(z(\rho)). \tag{55}$$

The map $\rho \mapsto z(\rho)$ is linear, so convexity of F makes $\mathcal{F}$ convex along linear mixtures of measures. Particle gradient flow converges in the mean-field limit to a 2-Wasserstein gradient flow (Ambrosio et al., 2008; Mei et al., 2018; Chizat & Bach, 2018). Since ϕ is nonlinear, $\mathcal{F}$ is generally not convex along Wasserstein geodesics. The following bound quantifies the resulting correction.

Define the Wasserstein gradient norm

$$\mathsf{G}(\rho) := \left(\int \left\|D\phi(u)^\top \nabla_z F(z(\rho))\right\|_2^2 \, d\rho(u)\right)^{1/2}. \tag{56}$$

Proposition 9 (Mean-field transport bound)**.** *Suppose that, along the transport segments under consideration,*

$$\left\|D^2\phi(u)[v,v]\right\|_2 \leq L_\phi \|v\|_2^2, \qquad \|\nabla_z F(z(\rho))\|_2 \leq N_F.$$

Let $\bar{\rho}$ and $\Delta \geq 0$ satisfy $\mathcal{F}(\bar{\rho}) - f^\star \leq \Delta$ and $W_2(\rho, \bar{\rho}) \leq R$ for some $R > 0$. Then

$$\mathsf{G}(\rho) \geq \frac{\mathcal{F}(\rho) - f^\star - \Delta}{R} - \frac{L_\phi N_F}{2} R. \tag{57}$$

Consequently, for any $a \in (0,1)$, if

$$\mathcal{F}(\rho) - f^\star \geq \frac{\Delta + \frac{1}{2}L_\phi N_F R^2}{1-a}, \tag{58}$$

then

$$\mathsf{G}(\rho) \geq \frac{a}{R}\big(\mathcal{F}(\rho) - f^\star\big). \tag{59}$$

Proof. Let π be an optimal coupling of ρ and $\bar{\rho}$. Convexity of F and a second-order expansion of ϕ at u give

$$\begin{aligned}
\mathcal{F}(\rho) - \mathcal{F}(\bar{\rho}) &\leq \langle \nabla_z F(z(\rho)), z(\rho) - z(\bar{\rho})\rangle \\
&\leq \int \left\langle D\phi(u)^\top \nabla_z F(z(\rho)), u - \bar{u}\right\rangle d\pi(u,\bar{u}) + \frac{L_\phi N_F}{2} W_2(\rho,\bar{\rho})^2 \\
&\leq \mathsf{G}(\rho) W_2(\rho,\bar{\rho}) + \frac{L_\phi N_F}{2} W_2(\rho,\bar{\rho})^2.
\end{aligned}$$

Adding $\mathcal{F}(\bar{\rho}) - f^\star \leq \Delta$ and using $W_2(\rho,\bar{\rho}) \leq R$ proves (57), and (59) follows from (58). □

The μ-KŁ guarantee above requires uniform control of the feature curvature, the logit-gradient norm and the loss gap, which limits its applicability. The curvature correction in (57) remains order one when neurons move by an order-one distance in the feature-learning regime. If the hidden features are fixed and only the linear readout parameters vary, then $L_\phi = 0$ and this correction vanishes, recovering the setting of Lemma 18.

Finite 2-layer networks. The comparator $\bar{\rho}$ may be the empirical neuron measure of any Δ-optimal finite network, and for empirical cross-entropy loss one may take $N_F = \sqrt{2/n}$. For a width-m two-layer network in the mean-field parameterization, let

$$\rho_x = \frac{1}{m}\sum_{r=1}^{m} \delta_{u_r}, \qquad f(x) = F\left(\frac{1}{m}\sum_{r=1}^{m} \phi(u_r)\right).$$

Then

$$\mathsf{G}(\rho_x) = \sqrt{m}\, \|\nabla_x f(x)\|_2 . \tag{60}$$

The proposition then yields

$$\sqrt{m}\, \|\nabla_x f(x)\|_2 \geq \frac{a}{R}\big(f(x) - f^\star\big).$$

Hence (59) yields the Euclidean constant $\mu = a/(R\sqrt{m})$. Equivalent parameterizations with different width-dependent scalar weight multipliers and layerwise learning rates can induce equivalent training dynamics in function space (Yang & Hu, 2021), but attain any other width scaling exponents in μ. This motivates formulating a geometry-aware μ-KŁ condition.

Width-independent μ requires considering preconditioned μ-KŁ. For the preconditioned gradient flow

$$\dot{x}(t) = -P\nabla f\big(x(t)\big), \qquad P \succeq 0,$$

the optimizer-matched μ-KŁ condition is

$$\sqrt{\nabla f(x)^\top P \nabla f(x)} \geq \mu_P\big(f(x) - f^\star\big).$$

The mean-field geometry corresponds to

$$\|v\|_{\mathrm{MF}}^2 = \frac{1}{m}\sum_{r=1}^{m} \|v_r\|_2^2, \; P = mI,$$

and hence to $\dot{x}(t) = -m\nabla_x f(x(t))$. In this geometry, the optimizer-matched constant a/R is width independent when $R = \Theta(1)$, as when individual neurons move an order-one distance, even though their concatenated Euclidean displacement is $\Theta(\sqrt{m})$. Thus the Euclidean decay $a/(R\sqrt{m})$ is a coordinate effect compensated by the mean-field learning-rate multiplier. For the scaling laws in the main text, more important than its absolute values is that the μ constant remains stable over the course of training.

For networks with three or more layers, mean-field limits can still describe the training dynamics, but the output depends nonlinearly on coupled layerwise measures, so the two-layer convexity-in-measure argument does not extend. Mean-field structure alone therefore yields neither global optimality nor a μ-KŁ bound. Such guarantees require additional assumptions, such as an NTK neighborhood, a well-conditioned Jacobian, or readout separability or margin conditions, as discussed in the other subsections.

F.4 The Lazy/NTK Regime

The general theorem of Bochnak & Łojasiewicz (1971) as well as the results of Sections F.1 and F.2 cover the behavior near stationarity. In this section we will try to characterize a complementary near-initialization regime, which the local theorems above do not apply. We now consider cross-entropy loss in the NTK regime. The guarantee reduces to a spectral condition on the data and holds with high probability over x_0. The following lemma lower bounds the loss-logit gradient in terms of the per-sample losses.

For x, define the minimal NTK eigenvalue restricted to the relevant logit subspace,

$$\lambda_{\mathcal{S}}(x) \coloneqq \inf_{u\in\mathcal{S},\, \|u\|_2=1} u^\top J(x)J(x)^\top u = \inf_{u\in\mathcal{S},\, \|u\|_2=1} \big\|J(x)^\top u\big\|_2^2, \quad J(x) \coloneqq \nabla_x z(x). \tag{61}$$

Lemma 19 (Cross-entropy logit gradient). *Suppose $\ell_i = \ell(z_i(x), y_i) \le M$ for every $i \in [n]$, and let $q_M \coloneqq {}^{(1-e^{-M})}/_{M}$, with $q_0 = 1$. Then,*

$$\|\nabla_z F(z(x))\|_2 \ge \frac{q_M}{\sqrt{n}} f(x) \ge \frac{q_M}{\sqrt{n}}\big(f(x) - f^\star\big). \tag{62}$$

Proof. The function $t \mapsto {}^{(1-e^{-t})}/_{t}$ is decreasing on $\mathbb{R}_+$. Hence, for every sample,

$$\|\nabla_z \ell(z_i, y_i)\|_2 \ge 1 - p_{i,y_i} = 1 - e^{-\ell_i} \ge q_M \ell_i.$$

Using the Cauchy–Schwarz inequality in the form $(\sum_i a_i^2)^{1/2} \ge n^{-1/2} \sum_i a_i$ for $a_i \ge 0$, we derive

$$\begin{aligned}
\|\nabla_z F(z(x))\|_2 &= \frac{1}{n}\Big(\sum_{i\in[n]} \|\nabla_z \ell(z_i, y_i)\|_2^2\Big)^{1/2} \\
&\ge \frac{1}{n\sqrt{n}} \sum_{i\in[n]} \|\nabla_z \ell(z_i, y_i)\|_2 \\
&\ge \frac{q_M}{n\sqrt{n}} \sum_{i\in[n]} \ell_i \\
&= \frac{q_M}{\sqrt{n}} f(x).
\end{aligned}$$

The last inequality in (62) uses $f^\star \ge 0$. □

Lemma 20 (NTK μ-KŁ along training). *Let $\mathcal{X} \subseteq \mathbb{R}^d$ and suppose that*

$$\lambda_{\mathcal{S}}(x_0) \ge \lambda_0 > 0, \qquad \sup_{x\in\mathcal{X}} \|J(x) - J(x_0)\|_{\mathrm{op}} \le \tfrac{1}{2}\sqrt{\lambda_0}, \qquad \sup_{i\in[n],\, x\in\mathcal{X}} \ell_i \le M.$$

Then, μ-KŁ holds on all of $\mathcal{X}$, and hence along any trajectory contained in $\mathcal{X}$, with $\mu = {}^{q_M\sqrt{\lambda_0}}/_{(2\sqrt{n})}$.

Proof. For every $u \in \mathcal{S}$ and $x \in \mathcal{X}$, $\big\|J(x)^\top u\big\|_2 \ge \big\|J(x_0)^\top u\big\|_2 - \|J(x) - J(x_0)\|_{\mathrm{op}} \|u\|_2 \ge \frac{1}{2}\sqrt{\lambda_0}\,\|u\|_2$, so $\lambda_{\mathcal{S}}(x) \ge {}^{\lambda_0}/_{4}$ throughout $\mathcal{X}$. Since $\nabla_z F \in \mathcal{S}$,

$$\|\nabla_x f(x)\|_2 = \big\|J(x)^\top \nabla_z F(z(x))\big\|_2 \ge \tfrac{1}{2}\sqrt{\lambda_0}\,\|\nabla_z F(z(x))\|_2 \ge \frac{q_M\sqrt{\lambda_0}}{2\sqrt{n}}\big(f(x) - f^\star\big),$$

where the last step is Lemma 19. □

For a wide enough two-layer network, both hypotheses hold. Consider

$$z_i(x) = \frac{1}{\sqrt{m}} A\,\sigma(W s_i), \qquad x = (W, A) \in \mathbb{R}^{m\times d_x} \times \mathbb{R}^{C\times m}, \tag{63}$$

with i.i.d. standard Gaussian entries at initialization x_0, inputs normalized to $\|s_i\|_2 = 1$, and an activation σ with $\sigma(0) = 0$, $\|\sigma'\|_\infty \le L_\sigma$ and β_σ-Lipschitz derivative σ'. Let

$$\Gamma^\infty_{ij} \coloneqq \mathbb{E}_{w\sim\mathcal{N}(0, I_{d_x})}\big[\sigma(w^\top s_i)\,\sigma(w^\top s_j)\big], \qquad \lambda_0 \coloneqq \lambda_{\min}(\Gamma^\infty), \tag{64}$$

so that λ_0 is a property of the data alone.

Theorem 10 (Explicit NTK μ-KŁ). *Assume $\lambda_0 \coloneqq \lambda_{\min}(\Gamma^\infty) > 0$ and fix $\delta \in (0,1)$ and $M > 0$. Set*

$$\Lambda \coloneqq (C+2)L_\sigma^2 + 4\beta_\sigma^2\Big(C + 2\log\frac{4m}{\delta}\Big), \qquad R \coloneqq \frac{1}{2}\sqrt{\frac{\lambda_0\, m}{2n\Lambda}},$$
$$\mathcal{X} \coloneqq \big\{x \mid \|x - x_0\|_2 \le R,\ \sup_{i\in[n]} \ell_i \le M\big\}.$$

Then there is an absolute constant $c_{\mathit{width}} > 0$ such that, whenever the width satisfies

$$m \ge c_{\mathit{width}}\, L_\sigma^4\, \frac{n^2}{\lambda_0^2}\, \log\frac{2n}{\delta}, \tag{65}$$

μ-KŁ holds on $\mathcal{X}$ with $\mu = q_M\sqrt{{\lambda_0}/{(8n)}}$, with probability at least $1 - \delta$ over the initialization x_0, where $q_M = (1 - e^{-M})/M$.

Proof. We denote by w_r the r row of W. Let $\varphi_r(s) \coloneqq \sigma(w_r^\top s)$ and we have:

$$\Gamma_{ij} \coloneqq \frac{1}{m} \sum_{r\in[m]} \varphi_r(s_i)\varphi_r(s_j)$$

Differentiating (63) with respect to A gives $\partial z_{i,c}/\partial A_{c'r} = \mathbf{1}[c = c']\varphi_r(s_i)/\sqrt{m}$, so the contribution of the readout block to $J(x)J(x)^\top$ is exactly $\Gamma \otimes I_C$ in the index ordering (i, c). The matrix $J_W J_W^T$ is positive semidefinite, hence:

$$J(x)J(x)^\top = J_A J_A^T + J_W J_W^T \succeq J_A J_A^T = \Gamma \otimes I_C$$

So:

$$\lambda_{\mathcal{S}}(x) \geq \lambda_{\min}(\Gamma(x)). \tag{66}$$

Now let us define the random variables $X_r^{ij} = \phi_r(s_i)\phi_r(s_j)$ and their centered versions $Y_r^{ij} = X_r^{ij} - \mathbb{E}\left[X_r^{ij}\right]$, which we will use to apply Bernstein's inequality. The requirements for which are that the random variables have mean zero, are independent and sub-exponential:

- *Mean Zero:* By definition the random variables Y_r^{ij} have mean zero.
- *Independent:* For different r the random variables since each variable X_r^{ij} depends only on the random varaible w_r and for different r, the w_r are independent Gaussian random variables.
- *sub-Exponential:* Using the techniques described in Vershynin (2019), we can show that Y_r^{ij} is $8L_\sigma^2$ sub-Exponential.

So since Y_r^{ij} satisfies the requirements we can apply Bernstein's inequality, and by definition $\frac{1}{m}\sum_{r=1}^m Y_r^{ij} = \frac{1}{m}\sum_{r=1}^m \left(X_r^{ij} - \mathbb{E}\left[X_r^{ij}\right]\right) = \Gamma_{ij} - \Gamma_{ij}^\infty$, so we have:

$$\mathbb{P}\left[\left|\Gamma_{ij} - \Gamma_{ij}^\infty\right|\right] \leq 2e^{-c_B m \min\left\{\frac{u^2}{64L_\sigma^4}, \frac{u}{8L_\sigma^2}\right\}} \tag{67}$$

For λ_0, it holds $\lambda_0 \leq \mathrm{Tr}(\Gamma^\infty)/n = \mathbb{E}\left[\sigma(Z^2)\right] \leq L_\sigma^2$, so by taking $u = \lambda_0/(2n)$ in Equation (67) the quadratic term is dominant. By taking a union bound over all i, j pairs, we have:

$$\mathbb{P}\left[\max_{i,j}\left|\Gamma_{ij}(x_0) - \Gamma_{ij}^\infty\right| \geq \frac{\lambda_0}{2n}\right] \leq 2n^2 e^{-\frac{c_B m \lambda_0^2}{64 n^2 L_\sigma^4}} \tag{68}$$

We set the right hand side of the above inequality to be at most $\delta/2$, which gives $m \geq \frac{64}{c_B} L_\sigma^4 \frac{n^2}{\lambda_0^2}\log(\frac{4n^2}{\delta}) = \frac{128}{c_B} L_\sigma^4 \frac{n^2}{\lambda_0^2}\log(\frac{2n}{\delta})$. So the event happens with probability at most $\delta/2$, for the m given in the theorem statement for $c_{\text{width}} = 128/c_B$.

For $\max_{i,j}\left|\Gamma_{ij}(x_0) - \Gamma_{ij}^\infty\right| \leq \lambda_0/(2n)$, we have that

$$\|\Gamma(x_0) - \Gamma^\infty\|_F^2 \leq n^2 \left(\max_{i,j}\left|\Gamma_{ij}(x_0) - \Gamma_{ij}^\infty\right|\right)^2 \leq n^2 \frac{\lambda_0^2}{4n^2} = \frac{\lambda_0^2}{4}$$

So $\|\Gamma(x_0) - \Gamma^\infty\|_{\text{op}} \leq \frac{\lambda_0}{2}$, which implies that $\lambda_{\min}(\Gamma(x_0)) \geq \lambda_0/2$. With (66) this implies that $\lambda_{\mathcal{S}}(x_0) \geq \lambda_0/2$.

For the Jacobian, let $a_{\max} \coloneqq \max_{r\in[m]}(\sum_{c\in[C]} A_{cr}(x_0)^2)^{1/2}$. A χ^2 tail bound and a union bound over $r \in [m]$ give $a_{\max}^2 \leq 2C + 4\log(4m/\delta)$ with probability at least $1 - \delta/2$. On this event, for any x with $\|x - x_0\|_2 \leq R$, bounding the readout and hidden blocks of $J(x) - J(x_0)$ separately and using $|\sigma(u) - \sigma(v)| \leq L_\sigma|u - v|$ and $|\sigma'(u) - \sigma'(v)| \leq \beta_\sigma|u - v|$,

$$\|J(x) - J(x_0)\|_{\text{op}}^2 \leq \|J(x) - J(x_0)\|_F^2 \leq \frac{n}{m}\Big[(C + 2)L_\sigma^2 + 2\beta_\sigma^2 a_{\max}^2\Big]R^2 \leq \frac{n\Lambda}{m}R^2,$$

so the choice of R yields $\|J(x) - J(x_0)\|_{\text{op}} \leq \frac{1}{2}\sqrt{\lambda_0/2}$. Both events hold simultaneously with probability at least $1-\delta$, and Lemma 20 applied with λ_0 replaced by $\lambda_0/2$ gives $\mu = q_M\sqrt{\lambda_0/2}/(2\sqrt{n}) = q_M\sqrt{\lambda_0/(8n)}$. □

The only hypothesis of Theorem 10 that is not directly checkable is $\lambda_0 > 0$. Expanding the activation in Hermite polynomials reduces it to a condition on the data alone.

Corollary 7 (μ-KŁ from a data assumption)**.** *Let the inputs be i.i.d. samples from a distribution on the unit sphere that is absolutely continuous with respect to the surface measure. Let* $c_k := \mathbb{E}_{Z\sim\mathcal{N}(0,1)}[\sigma(Z)h_k(Z)]$ *denote the Hermite coefficients of* σ*, with* h_k *the* k*-th normalized Hermite polynomial, and fix an order* k *with* $c_k \neq 0$ *and*

$$n \leq \binom{d_x + k - 1}{k}. \tag{69}$$

Write $G := SS^\top$ *for the input Gram matrix,* $G^{\circ k}$ *for its* k*-fold entrywise power,* $g := \lambda_{\min}(G^{\circ k})$*, and* $\lambda_k := c_k^2 g$*. Define*

$$R_k := \frac{1}{2}\sqrt{\frac{\lambda_k m}{2n\Lambda}}, \qquad \mathcal{X}_k := \big\{x \mid \|x - x_0\|_2 \leq R_k,\ \sup_{i\in[n]} \ell_i \leq M\big\},$$

where Λ *and* M *are as in Theorem 10. Then, for every* $\delta \in (0,1)$ *and every width*

$$m \geq c_0 L_\sigma^4 n^2 (c_k^2 g)^{-2} \log n/\delta,$$

μ-KŁ holds on $\mathcal{X}_k$ *with*

$$\mu = q_M\, |c_k| \sqrt{\frac{g}{8n}},$$

for almost every sampled data set, with probability at least $1 - \delta$ *over the initialization.*

Proof. The standing assumptions $\sigma(0) = 0$ and $\|\sigma'\|_\infty \leq L_\sigma$ give $|\sigma(t)| \leq L_\sigma|t|$, so $\sigma \in L^2(\mathcal{N}(0,1))$ and its Hermite series converges. For unit vectors s_i, s_j and $w \sim \mathcal{N}(0, I_{d_x})$ the pair $(w^\top s_i, w^\top s_j)$ is centered Gaussian with unit variances and correlation G_{ij}, so $\mathbb{E}[h_k(u)h_l(v)] = \mathbf{1}[k = l]\, G_{ij}^k$ and expanding $\sigma = \sum_{k\geq 0} c_k h_k$ gives $\Gamma^\infty = \sum_{k\geq 0} c_k^2 G^{\circ k}$. Every $G^{\circ k}$ is positive semidefinite by the Schur product theorem, hence $\Gamma^\infty \succeq c_k^2 G^{\circ k}$ for each single k and $\lambda_0 \geq c_k^2 g$.

It remains to check $g > 0$ almost surely. Since $G_{ij}^{\circ k} = \langle s_i, s_j\rangle^k = \langle s_i^{\otimes k}, s_j^{\otimes k}\rangle$, the matrix $G^{\circ k}$ is the Gram matrix of the symmetric tensor powers $s_i^{\otimes k}$, which lie in a space of dimension $\binom{d_x+k-1}{k}$. For inputs drawn from an absolutely continuous distribution these n tensors are linearly independent almost surely whenever n does not exceed that dimension, which is (69), so $G^{\circ k} \succ 0$ almost surely. Substituting $\lambda_0 \geq c_k^2 g$ into Theorem 10 gives the claim. □

The order k controls how many samples the positivity argument tolerates. Taking $k = 1$ requires linearly independent inputs and hence $n \leq d_x$, while $k = 2$ allows n of order d_x^2 and order k allows n of order d_x^k. This is an almost-sure positivity statement only: absolute continuity and the dimension bound do not provide a deterministic lower bound on g, which can be arbitrarily small for poorly conditioned samples.

The constant μ does not improve with width, as it is set by the data through λ_0 and decaying as $n^{-1/2}$.[3] The effective benign radius also stays width-independent. To see this note that Condition (65) brings the kernel at initialization close to its infinite-width limit, and the radius R grows like $\sqrt{m/n}$ up to logarithmic factors. But the network (63) has $m(d_x + C)$ parameters, so $\|x_0\|_2 = \Theta(\sqrt{m})$ and the relative radius $R/\|x_0\|_2 = \Theta(n^{-1/2})$ is width-independent.

Theorem 10 is a cross-entropy analogue of the PŁ* bounds of Liu et al. (2022), with the Jacobian perturbation of Oymak & Soltanolkotabi (2020). For ReLU the limiting kernel is the degree-one arc-cosine kernel (Cho & Saul, 2009; Jacot et al., 2018) and sharper lower bounds on λ_0 are available (Nguyen et al., 2021; Karhadkar et al., 2024), but ReLU is excluded by the hypothesis $\beta_\sigma < \infty$, which is what makes the radius positive. Whether the iterates in fact remain in $\mathcal{X}$ is a separate question, answered for overparameterized deep ReLU networks under SGD by Zou et al. (2018). The guarantee applies throughout the portion of the NTK ball on which the per-sample losses remain bounded by M where $M \geq \max_i \ell_i(x_0)$.

[3] M is finite on any bounded $\mathcal{X}$, since $\ell_i \leq \log C + 2\|z_i\|_\infty$, and it enters only through q_M.

To conclude, the three settings highlight complementary regimes in which μ-KŁ can be guaranteed for a specified region. The linear model gives μ in closed form on a low-loss region, using an imbalance that gradient flow conserves. The readout analysis provides guarantees in various regimes in which the learned features are rich enough, once the global optimum is covered by the features, the current logits have a uniform positive margin or the features admit a controlled separator. Finally, the lazy-regime analysis keeps μ stable near initialization under a data-spectrum condition and bounded per-example losses.

G Lower Bounds for Stochastic Methods under μ-KŁ

In this section, we present lower bounds which show that the convergence rates obtained earlier are optimal in their dependence on the target accuracy, the Lipschitz constant of the gradient L, and the noise level σ^2. This is achieved by constructing a hard instance, i.e., a Lipschitz-smooth nonconvex function satisfying the μ-KŁ inequality for which no *zero-respecting* stochastic method can achieve better than ε optimization error in less than $N = \Omega\left(L/\varepsilon + L\sigma^2/\varepsilon^3\right)$ oracle calls (see Theorem 11 for the precise statement). The class of zero-respecting algorithms (Definition G.4) includes `SGD`, `SGD` with momentum, and `AdamW`, whose updates act coordinate-wise on the observed gradients. The construction adapts the *zero-chain* property (Definition G.1) underlying the deterministic lower bounds of Nesterov (2004); Carmon et al. (2019) and their stochastic extension by Arjevani et al. (2023) to the μ-KŁ class.

We remind that d, T, B are used to refer to the dimension of the problem, token budget, and batch size B, respectively. In this section, we work with the Euclidean norm only. Thus, we omit the subscript 2 for simplicity, i.e., $\|\cdot\|$ denotes the Euclidean norm. Superscripts index oracle queries and subscript index coordinates, so x_i^k is the i^{th} coordinate of the k-th query.

G.1 Definition of Zero-chains

The construction is based on a function whose gradient can be nonzero on at most one new coordinate at each step. Then, any algorithm which is *zero-respecting* (Definition G.4), i.e., whose iterates stay inside the coordinates it has already observed, must query the oracle at least once per coordinate before it can approach the minimizer.

Definition G.1 (Zero-chain). *A function $F : \mathbb{R}^d \to \mathbb{R}$ is a* zero-chain with threshold $\tau \geq 0$*, if for every $x \in \mathbb{R}^d$,*

$$\operatorname{supp}\left(\nabla F(x)\right) \subseteq \{1, \ldots, \operatorname{prog}_\tau(x) + 1\},$$

where

$$\operatorname{prog}_\tau(x) \coloneqq \max\{i \in \{1, \ldots, d\} : |x^i| > \tau\} \tag{70}$$

with a convention $\operatorname{prog}_\tau(x) \coloneqq 0$ if no such i exists.

The quantity $\operatorname{prog}_\tau(x)$ is the largest activated coordinate index. By design, the gradient of a zero-chain vanishes identically beyond coordinate $\operatorname{prog}_\tau(x)+1$. The zero-chain property can be extended to the stochastic setting in the following way.

Definition G.2 (Stochastic zero-chain). *Let F be a zero-chain with threshold τ (Definition G.1) and let $\sigma \geq 0$. A stochastic oracle g* realizes F as a stochastic zero-chain with noise level σ and reveal probability $\pi \in (0, 1]$ *if for $x \in \mathbb{R}^d$ with $\operatorname{prog}_\tau(x) < d$, we have*

$$\begin{aligned}
&\textbf{\textit{Unbiasedness:}} && \mathbb{E}[g(x) \mid x] = \nabla F(x);\\
&\textbf{\textit{Bounded variance:}} && \mathbb{E}\left[\|g(x) - \nabla F(x)\|^2 \mid x\right] \leq \sigma^2;\\
&\textbf{\textit{Chance of revealing the frontier:}} && \exists \zeta \sim \text{Bernoulli}(\pi) \quad \textit{such that}\\
& && \textit{if } \zeta = 0, \textit{ then } \operatorname{supp}(g(x)) \subseteq \{1, \ldots, \operatorname{prog}_\tau(x)\}\\
& && \textit{if } \zeta = 1, \textit{ then } \operatorname{supp}(g(x)) \subseteq \{1, \ldots, \operatorname{prog}_\tau(x) + 1\}.
\end{aligned}$$

G.2 Scalar Building Blocks

In this section, we provide technical lemmas and estimates that are used in the construction of the zero-chain H_d, which will be the basis for the lower bound. For a given integer $d \geq 2$, let $q \coloneqq 16d$.

We define the following real-valued functions, which we call the smooth cutoff ω, its rescaling ψ_q, and the gain function ϕ_q,

$$\chi(s) \coloneqq \begin{cases} e^{-1/s} & s > 0 \\ 0 & s \le 0 \end{cases}, \qquad \omega(s) \coloneqq \frac{\chi(s)}{\chi(s) + \chi(1-s)}, \tag{71}$$

$$\psi_q(t) \coloneqq \omega(2qt - 1), \qquad \phi_q(t) \coloneqq \frac{1 - (1-t)^q}{1 + (1-t)^q}. \tag{72}$$

Lemma 21 (Properties of ω and ψ_q)**.** *Let ω and ψ_q be defined as in* (71) *and* (72)*, respectively. Then, for ψ_q and ω we have*

$$\psi_q(t) \in [0,1] \quad \textit{for all } t, \quad \psi_q(t) = \begin{cases} 0 & t \le \frac{1}{2q} \\ 1 & t \ge \frac{1}{q} \end{cases}, \quad \psi_q'(t) \ge 0 \tag{73}$$

$$\psi_q'(t) = 0 \quad \textit{for all} \quad t \le \tfrac{1}{2q} \textit{ and } t \ge \tfrac{1}{q},$$

and

$$\|\omega'\|_\infty = 2, \quad \|\omega''\|_\infty \le 108, \quad |\psi_q'(t)| \le 4q \quad \textit{and} \quad |\psi_q''(t)| \le 432q^2 \quad \forall t \in \mathbb{R}. \tag{74}$$

Proof. Note that we have $\chi \in C^\infty(\mathbb{R}), \chi(s) \ge 0$, and $\chi(s) + \chi(1-s) > 0$ for all $s \in \mathbb{R}$. This implies that $\omega \in C^\infty(\mathbb{R})$ and $\omega(s) \in [0,1]$ for all $s \in \mathbb{R}$. Next, since $\chi(s) = 0$ for all $s \le 0$, then the same holds for ω: $\omega(s) = 0$ for all $s \le 0$ by the definition. Moreover, note that for $s \ge 1$, we have $\chi(1-s) = 0$. This implies that $\omega(s) = \frac{\chi(s)}{\chi(s)} = 1$ for all $s \ge 1$.

Knowing the properties of ω, we now transfer them to $\psi_q(t) = \omega(2qt-1)$. First, since $\omega(s) \in [0,1]$ for all $s \in \mathbb{R}$, then the same holds for $\psi_q(t) \in [0,1]$ for all $t \in \mathbb{R}$. For $t \le \frac{1}{2q}$, we have $2qt - 1 \le 0$. Therefore, $\psi_q(t) = 0$. Similarly, for $t \ge \frac{1}{q}$, we have $2qt - 1 \ge 1$, thus $\psi_q(t) = 1$.

Now we compute the derivative of $\omega(s)$

$$\omega'(s) = \frac{\chi'(s)\chi(1-s) + \chi(s)\chi'(1-s)}{(\chi(s) + \chi(1-s))^2}.$$

Note that $\chi(s) = \chi'(s) = 0$ for $s \le 0$. This implies that $\omega'(s) = 0$ for all $s \le 0$ and $s \ge 1$. Consequently, $\psi_q'(t) = 0$ for all $t \le \frac{1}{2q}$ and $t \ge \frac{1}{q}$.

It remains to prove the derivative bounds. Since ω is constant outside $(0,1)$, it suffices to consider $s \in (0,1)$. For simplicity of calculations, we define functions

$$h(s) \coloneqq \frac{1}{s^2} + \frac{1}{(1-s)^2}, \quad g(s) \coloneqq \frac{1}{s} - \frac{1}{1-s}.$$

For $s \in (0,1)$, the definition of ω can be written as

$$\omega(s) = \frac{e^{-1/s}}{e^{-1/s} + e^{-1/(1-s)}} = \frac{1}{1 + \exp(1/s - 1/1-s)} = \frac{1}{1 + e^{g(s)}}.$$

Since $g'(s) = -h(s)$, differentiating $\omega(s)$ gives

$$\omega'(s) = \frac{e^{g(s)}}{(1 + e^{g(s)})^2} h(s) = \omega(s)(1 - \omega(s))h(s).$$

Now we estimate ω'. Let $r \coloneqq s(1-s)$, so that $0 < r \le \frac{1}{4}$. A direct computation yields

$$h(s) = \frac{1 - 2r}{r^2}, \quad g(s)^2 = \frac{1 - 4r}{r^2}.$$

Moreover,

$$\omega(s)(1 - \omega(s)) = \frac{e^{g(s)}}{(1 + e^{g(s)})^2} = \frac{1}{4\cosh^2(g(s)/2)}.$$

Since $\cosh^2(x) \geq 1 + x^2$ for all $x \in \mathbb{R}$, we have

$$\cosh^2\left(\frac{g(s)}{2}\right) \geq 1 + \frac{g(s)^2}{4} = \frac{(1-2r)^2}{4r^2}.$$

Therefore,

$$0 \leq \omega'(s) = \omega(s)(1-\omega(s))h(s) = \frac{1}{4\cosh^2(g(s)/2)} \cdot \frac{1-2r}{r^2}$$
$$= \frac{1-2r}{4r^2\cosh^2(g(s)/2)} \leq \frac{1-2r}{4r^2} \cdot \frac{4r^2}{(1-2r)^2} \leq \frac{1}{1-2r} \leq 2.$$

At $s = \frac{1}{2}$, we have $\omega\left(1/2\right) = \frac{1}{2}, h\left(1/2\right) = 8$, and hence $\omega'\left(1/2\right) = \frac{1}{4} \cdot 8 = 2$. Since $\omega' = 0$ outside $(0,1)$, it follows that $\|\omega'\|_\infty = 2$. Moreover, the bound $0 \leq \omega'$ implies that $\psi_q' \geq 0$ as well.

Next, we estimate ω''. Differentiating $\omega'(s) = \omega(s)(1-\omega(s))h(s)$, we obtain

$$\omega''(s) = \omega(s)(1-\omega(s))\left(h'(s) + (1-2\omega(s))h(s)^2\right),$$

where $h'(s) = -\frac{2}{s^3} + \frac{2}{(1-s)^3}$. Since $|1-2\omega(s)| \leq 1$, we obtain the bound

$$|\omega''(s)| \leq \omega(s)(1-\omega(s))\left(|h'(s)| + h(s)^2\right).$$

We consider three cases for $s \in (0,1)$. First, let $s \in [1/3, 2/3]$, then $h(s) \leq 18$ and

$$|h'(s)| \leq \frac{2}{s^3} + \frac{2}{(1-s)^3} \leq 108.$$

Since $\omega(s)(1-\omega(s)) \leq \frac{1}{4}$, it follows that

$$|\omega''(s)| \leq \frac{1}{4}\left(108 + 18^2\right) = 108.$$

Second, let $s \in (0, 1/3]$, and set $y := 1/s \geq 3$. Since $(1-s)^{-1} \leq 3/2$, we have

$$g(s) = \frac{1}{s} - \frac{1}{1-s} \geq y - \frac{3}{2} > 0.$$

Therefore,

$$\omega(s)(1-\omega(s)) = \frac{e^{g(s)}}{(1+e^{g(s)})^2} \leq e^{-g(s)} \leq e^{3/2}e^{-y}.$$

Furthermore, $h(s) \leq y^2 + \frac{9}{4} \leq y^2 + \frac{y^2}{4} \leq \frac{5}{4}y^2$, and

$$|h'(s)| \leq 2y^3 + \frac{27}{4} \leq \frac{9}{4}y^3 \leq \frac{3}{4}y^4.$$

Hence,

$$|h'(s)| + h(s)^2 \leq \left(\frac{3}{4} + \frac{25}{16}\right)y^4 = \frac{37}{16}y^4.$$

The function $y \mapsto y^4 e^{-y}$ attains its maximum on $[3,\infty)$ at $y = 4$. Thus,

$$|\omega''(s)| \leq \frac{37}{16}e^{3/2} \cdot 4^4 e^{-4} = 592e^{-5/2} < 108.$$

Finally, the identity $\omega(1-s) = 1 - \omega(s)$ implies, after differentiating twice, that

$$\omega''(1-s) = -\omega''(s).$$

Thus, the same estimate holds for $s \in [\frac{2}{3}, 1)$. Since $\omega'' = 0$ outside $(0,1)$, we conclude that $\|\omega''\|_\infty \leq 108$ for all $s \in \mathbb{R}$. Knowing the bounds for the derivatives of ω, using the chain rule, we transfer them to the bounds on the derivatives of ψ_q

$$\psi_q'(t) = 2q\omega'(2qt-1), \quad \psi_q''(t) = 4q^2\omega''(2qt-1).$$

Consequently, for all $t \in \mathbb{R}$,

$$|\psi_q'(t)| \leq 2q\|\omega'\|_\infty = 4q, \quad \text{and} \quad |\psi_q''(t)| \leq 4q^2\|\omega''\|_\infty \leq 432q^2.$$

□

Lemma 22 (Properties of ϕ_q)**.** *Let ϕ_q be defined as in* (72)*. Then, we have that*

(i) *endpoint values and the range of possible values of ϕ_q are*

$$\phi_q(0) = 0, \quad \phi_q(1) = 1, \quad 0 \le \phi_q(t) \le 1 \quad \textit{for all } t \in [0,1], \quad |\phi_q(t)| \le 1 \quad \textit{for all } t \in \mathbb{R}; \tag{75}$$

(ii) *global derivative bounds of $\phi_q'(t)$ are*

$$|\phi_q'(t)| \le 2q \quad \textit{and} \quad |\phi_q''(t)| \le 6q^2 \quad \textit{for all } t \in \mathbb{R}; \tag{76}$$

(iii) *ϕ_q satisfies KŁ inequality of the form*

$$1 - \phi_q(t) \le \frac{2}{q}|\phi_q'(t)| \quad \textit{for all } t \in [0,2]; \tag{77}$$

(iv) *derivative $\phi'(t)$ is lower bounded as follows*

$$\phi_q'(t) \ge \frac{q}{8} \quad \textit{for all } t \in \left[0, \frac{1}{q}\right]; \tag{78}$$

(v) *the second-order derivative of ϕ_q at zero is*

$$\phi_q''(0) = \frac{q}{2} > 0. \tag{79}$$

Proof.
Endpoint values and range. By the definition of ϕ_q, we have

$$\phi_q(0) = \frac{1-(1-0)^q}{1+(1-0)^q} = 0, \quad \phi_q(1) = \frac{1-(1-1)^q}{1+(1-1)^q} = 1.$$

Let $t \in [0,1]$, then $0 \le 1-t \le 1$, and therefore $0 \le (1-t)^q \le 1$. This implies that

$$0 \le 1-(1-t)^q \le 1+(1-t)^q.$$

Since $1+(1-t)^q > 0$, dividing by this quantity gives $0 \le \phi_q(t) \le 1$ for all $t \in [0,1]$. More generally, since q is even, $(1-t)^q \ge 0$ for all $t \in \mathbb{R}$. Hence, we derive

$$|1-(1-t)^q| \le 1+(1-t)^q.$$

Dividing by the positive quantity $1+(1-t)^q$, we obtain

$$|\phi_q(t)| = \frac{|1-(1-t)^q|}{1+(1-t)^q} \le 1 \quad \text{for all } t \in \mathbb{R}.$$

Global derivative bounds. Let $u = 1-t$. By the quotient rule, we get

$$\phi_q'(t) = \frac{2qu^{q-1}}{(1+u^q)^2}.$$

Since q is even, we have $u^q = |u|^q$, and

$$\left|\phi_q'(t)\right| = \frac{2q\,|u|^{q-1}}{(1+|u|^q)^2}.$$

Next, assume that $|u| \le 1$. Then

$$|u|^{q-1} \le 1, \quad \text{and} \quad (1+|u|^q)^2 \ge 1.$$

Consequently, $\left|\phi_q'(t)\right| \le 2q$.

Now, assume that $|u| \ge 1$. Then we have $(1+|u|^q)^2 \ge |u|^{2q}$. This implies that

$$\left|\phi_q'(t)\right| \le \frac{2q\,|u|^{q-1}}{|u|^{2q}} = \frac{2q}{|u|^{q+1}} \le 2q.$$

Therefore, we obtain that the first-order derivative is bounded in both cases as $\left|\phi_q'(t)\right| \leq 2q$ for all $t \in \mathbb{R}$.

Differentiating $\phi_q(t)$ once more gives

$$\phi_q''(t) = -\frac{2q(q-1)u^{q-2}}{(1+u^q)^2} + \frac{4q^2u^{2q-2}}{(1+u^q)^3}.$$

Taking absolute values and using $u^q = |u|^q$, we obtain

$$\left|\phi_q''(t)\right| \leq \frac{2q(q-1)\,|u|^{q-2}}{(1+|u|^q)^2} + \frac{4q^2\,|u|^{2q-2}}{(1+|u|^q)^3}.$$

Again, if we assume that $|u| \leq 1$, then

$$|u|^{q-2} \leq 1 \quad \text{and} \quad |u|^{2q-2} \leq 1.$$

Hence, we get

$$\left|\phi_q''(t)\right| \leq 2q(q-1) + 4q^2 \leq 6q^2.$$

If we assume that $|u| \geq 1$, then we have

$$(1+|u|^q)^2 \geq |u|^{2q}$$

and

$$(1+|u|^q)^3 \geq |u|^{3q}.$$

Therefore, we derive that

$$\frac{|u|^{q-2}}{(1+|u|^q)^2} \leq \frac{|u|^{q-2}}{|u|^{2q}} = \frac{1}{|u|^{q+2}} \leq 1,$$

and similarly,

$$\frac{|u|^{2q-2}}{(1+|u|^q)^3} \leq \frac{|u|^{2q-2}}{|u|^{3q}} = \frac{1}{|u|^{q+2}} \leq 1.$$

This implies that

$$\left|\phi_q''(t)\right| \leq 2q(q-1) + 4q^2 \leq 6q^2.$$

Thus, we obtain that the second-order derivative is bounded in both cases as

$$\left|\phi_q''(t)\right| \leq 6q^2 \quad \text{for all } t \in \mathbb{R}.$$

KŁ inequality. Let $t \in [0,2]$. Then, we have $|1-t| \leq 1$. Since q is even, we also get $(1-t)^q = |1-t|^q$. Therefore,

$$1 - \phi_q(t) = 1 - \frac{1-|1-t|^q}{1+|1-t|^q} = \frac{2\,|1-t|^q}{1+|1-t|^q}, \quad \text{and} \quad \left|\phi_q'(t)\right| = \frac{2q\,|1-t|^{q-1}}{(1+|1-t|^q)^2}.$$

If $t = 1$, then we trivially have

$$1 - \phi_q(t) = 0 \quad \text{and} \quad \left|\phi_q'(t)\right| = 0,$$

so the desired inequality holds.

Now, we assume that $t \neq 1$. Dividing the previous equations gives us

$$\frac{1-\phi_q(t)}{\left|\phi_q'(t)\right|} = \frac{|1-t|\,(1+|1-t|^q)}{q}.$$

Since $|1-t| \leq 1$ and $|1-t|^q \leq 1$, we have

$$|1-t|\,(1+|1-t|^q) \leq 2.$$

Consequently,

$$\frac{1-\phi_q(t)}{\left|\phi_q'(t)\right|} \leq \frac{2}{q}.$$

Equivalently,

$$1 - \phi_q(t) \le \frac{2}{q} \left|\phi_q'(t)\right|.$$

Derivative lower bound. Let $t \in [0, 1/q]$. Then

$$1 - \frac{1}{q} \le 1 - t \le 1.$$

In particular, $1 - t \ge 0$, and hence

$$\phi_q'(t) = \frac{2q(1-t)^{q-1}}{\left(1 + (1-t)^q\right)^2}.$$

Since $1 - t \le 1$, we have $(1-t)^q \le 1$, and therefore $\left(1 + (1-t)^q\right)^2 \le 4$. Moreover, since $1 - t \ge 1 - \frac{1}{q}$, we have

$$(1-t)^{q-1} \ge \left(1 - \frac{1}{q}\right)^{q-1}.$$

Using the standard estimate

$$\left(1 + \frac{1}{n}\right)^n \le e \quad \text{for all } n \ge 1$$

with $n = q - 1$, we obtain

$$\left(1 - \frac{1}{q}\right)^{q-1} = \left(1 + \frac{1}{q-1}\right)^{-(q-1)} \ge e^{-1} > \frac{1}{4}.$$

This implies that

$$\phi_q'(t) \ge \frac{2q \cdot \frac{1}{4}}{4} = \frac{q}{8}.$$

Second-order derivative at zero. The formula for the second derivative can be rewritten as follows

$$\phi_q''(t) = \frac{2q(1-t)^{q-2}\left((q+1)(1-t)^q - (q-1)\right)}{\left(1 + (1-t)^q\right)^3}.$$

Evaluating it at $t = 0$, we obtain

$$\phi_q''(0) = \frac{2q\left((q+1) - (q-1)\right)}{(1+1)^3} = \frac{2q \cdot 2}{8} = \frac{q}{2}.$$

Since $q > 0$, this implies that $\phi_q''(0) = \frac{q}{2} > 0$. □

Next, we construct the hard instance function H_d that is simultaneously smooth, nonconvex, μ-KŁ, and a zero-chain, as summarized in Lemma 27. We start by constructing an auxiliary function $\overline{H}_d$ from the scalar building blocks of Section G.2. We fix $x^0 := 1$. Using the rescaled cutoff ψ_q and the gain ϕ_q from (72), we define

$$\overline{H}_d(x) := d - \psi_q(x^0)\phi_q(x^1) - \sum_{i=2}^{d} \psi_q(x^{i-1})\phi_q(x^i) \quad \text{for all} \quad x \in \mathbb{R}^d, \tag{80}$$

where $x^0 = 1$ is a fixed parameter and $x = (x^1, \ldots, x^d) \in \mathbb{R}^d$ is the argument of $\overline{H}$.

Lemma 23 (Smoothness of $\overline{H}_d$)**.** *Let $\overline{H}_d$ be defined as in* (80)*. Then, for all $x \in \mathbb{R}^d$ we have*

$$\left\|\nabla^2 \overline{H}_d(x)\right\|_{\mathrm{op}} \le 454\, q^2.$$

This implies that $\overline{H}_d$ is $454q^2$-smooth on $\mathbb{R}^d$.

Proof. Direct differentiation of (80) gives, for $i < d$,

$$\frac{\partial \overline{H}_d(x)}{\partial x^i} = -\psi_q(x^{i-1})\phi_q'(x^i) - \psi_q'(x^i)\phi_q(x^{i+1}),$$

and $\frac{\partial}{\partial x^d}\overline{H}_d(x) = -\psi_q(x^{d-1})\phi_q'(x^d)$. Since each summand $\psi_q(x^{i-1})\phi_q(x^i)$ depends only on neighboring coordinates, the Hessian of $\overline{H}_d$ is tri-diagonal. The diagonal entries satisfy

$$\left|\frac{\partial^2}{(\partial x^i)^2}\overline{H}_d(x)\right| \le |\phi_q''(x^i)| + |\psi_q''(x^i)||\phi_q(x^{i+1})| \le 6q^2 + 432q^2 = 438q^2,$$

where $|\phi_q''(x^i)| \le 6q^2$ and $|\phi_q(x^{i+1})| \le 1$ follow from (75) and (76) of Lemma 22, and $|\psi_q''(x^i)| \le 432q^2$ follows from (74) of Lemma 21. The off-diagonal entries satisfy

$$\left|\frac{\partial^2}{\partial x^i \partial x^{i-1}}\overline{H}_d(x)\right| \le |\psi_q'(x^{i-1})||\phi_q'(x^i)| \le 4q \cdot 2q = 8q^2,$$

and

$$\left|\frac{\partial^2}{\partial x^i \partial x^{i+1}}\overline{H}_d(x)\right| \le |\psi_q'(x^i)||\phi_q'(x^{i+1})| \le 4q \cdot 2q = 8q^2,$$

where $|\psi_q'| \le 4q$ is from (74) of Lemma 21 and $|\phi_q'| \le 2q$ is from (76) of Lemma 22. Note that $\frac{\partial^2}{\partial x^i \partial x^{i+1}}\overline{H}_d(x)$ makes sense only for $i > 1$, since $x^0 = 1$ is a fixed constant. Each row has at most two nonzero off-diagonal entries, so the row-sum bound for symmetric matrices gives

$$\left\|\nabla^2 \overline{H}_d(x)\right\|_{\mathrm{op}} \le 438q^2 + 2 \cdot 8q^2 = 454q^2.$$

□

Lemma 24 (Global minimizer of $\overline{H}_d$)**.** *Let $\overline{H}_d$ be defined as in* (80)*. Then, it satisfies*

$$\overline{H}_d(x) \ge 0 \quad \textit{for all} \quad x \in \mathbb{R}^d, \quad \textit{and} \quad \overline{H}_d(x^\star) = 0 \quad \textit{at} \quad x^\star \coloneqq (1, \dots, 1) \in \mathbb{R}^d,$$

i.e., $x^\star$ is the global minimizer. Moreover, $x^\star$ is unique.

Proof. Since q is even, $(1-t)^q \ge 0$ for all $t \in \mathbb{R}$, so $\phi_q(t) \le 1$ globally. Combined with $0 \le \psi_q(t) \le 1$ from Lemma 21, each term satisfies $\psi_q(x^{i-1})\phi_q(x^i) \le 1$, hence

$$\overline{H}_d(x) = d - \sum_{i=1}^{d} \psi_q(x^{i-1})\phi_q(x^i) \ge 0.$$

At $x^\star$ we have $(x^\star)^i = 1$ for all i. Moreover, from Lemmas 21 and 22, we have $\psi_q(1) = 1$ and $\phi_q(1) = 1$, so every summand equals one. This implies that

$$\overline{H}_d(x^\star) = d - \sum_{i=1}^{d} 1 = 0,$$

confirming that $x^\star$ is a global minimizer. Uniqueness follows because equality $\overline{H}_d(x^\star) = 0$ requires each summand to be

$$\psi_q((x^\star)^{i-1})\phi_q((x^\star)^i) = 1.$$

Since we have fixed $x^0 = 1$, then $\psi_q(x^0) = 1$ by Lemma 21. Therefore, we should have $\phi_q((x^\star)^1) = 1$ as well. From Lemma 22, this is possible only when $(x^\star)^1 = 1$. Continuing this argument, we arrive at $(x^\star)^i = 1$ for all $i \in [d]$. Thus, $x^\star = (1, 1, \dots, 1) \in \mathbb{R}^d$ is a unique minimizer. □

Lemma 25 (KŁ-type inequality for $\overline{H}_d$)**.** *Let $\overline{H}_d$ be defined as in* (80)*. Then, $\overline{H}_d$ satisfies μ-KŁ condition with $\mu = 1$ in the box $[0,2]^d$, namely, for all $x \in [0,2]^d$ we have*

$$\left\|\nabla \overline{H}_d(x)\right\| \ge \overline{H}_d(x) - \overline{H}_d(x^\star).$$

Proof. From Lemma 24, we have $\overline{H}_d(x^\star) = 0$. Therefore, it suffices to show $\left\|\nabla \overline{H}_d(x)\right\| \geq \overline{H}_d(x)$. Next, for $t \in [0,2]$ we have $|1-t| \leq 1$ and $(1-t)^q = |1-t|^q \leq 1$. This implies that for $t \in [0,2]$, we have $0 \leq \phi_q(t) \leq 1$. From Lemma 21, we also have $\psi_q \geq 0$. Therefore, we have $0 \leq \overline{H}_d(x) \leq d$.

To show that μ-KŁ inequality holds for $\overline{H}_d$, we consider two possible cases.

Case 1: some coordinate below $1/q$. Let ℓ be the smallest index such that $x^\ell < 1/q$. Assume that $\ell < d$. Thus, for every $1 \leq j < \ell$, we have $x^j \geq 1/q$. By Lemma 21, we obtain that $\psi_q(x^{\ell-1}) = 1$. Besides, since $x^0 = 1$, then $\psi_q(x^0) = 1$ as well. Since $x^\ell < 1/q < 1$, then by Lemma 22 we have $\phi_q'(x^\ell) \geq q/8 > 0$. Differentiating (80) with respect to x^ℓ and using $\psi_q(x^{\ell-1}) = 1$ gives

$$\begin{aligned} -\frac{\partial \overline{H}_d(x)}{\partial x^\ell} &= \psi_q(x^{\ell-1})\phi_q'(x^\ell) + \psi_q'(x^\ell)\phi_q(x^{\ell+1}) \\ &= \phi_q'(x^\ell) + \psi_q'(x^\ell)\phi_q(x^{\ell+1}) \geq \phi_q'(x^\ell), \end{aligned}$$

where $\psi_q'(x^\ell)\phi_q(x^{\ell+1}) \geq 0$ on $[0,2]^d$. By (78), $\phi_q'(x^\ell) \geq q/8$, so

$$\left\|\nabla \overline{H}_d(x)\right\| \geq \left|\frac{\partial}{\partial x^\ell}\overline{H}_d(x)\right| \geq \frac{q}{8} = 2d \geq \overline{H}_d(x),$$

where the last step uses $\overline{H}_d(x) \leq d$ and $q = 16d$. If $\ell = d$, then

$$-\frac{\partial \overline{H}_d(x)}{\partial x^\ell} = \psi_q(x^{\ell-1})\phi_q'(x^\ell) = \phi_q'(x^\ell),$$

and the rest of the proof follows the same reasoning as above.

Case 2: every coordinate at least $1/q$. In this case, by Lemma 21, we have that $\psi_q(x^i) = 1$ and $\psi_q'(x^i) = 0$ for all $i \in [d]$, so $\frac{\partial}{\partial x^i}\overline{H}_d(x) = -\phi_q'(x^i)$ and $\overline{H}_d(x) = \sum_{i=1}^d (1 - \phi_q(x^i))$. Applying (77) coordinate-wise and then using Cauchy-Schwarz inequality, we get

$$\overline{H}_d(x) = \sum_{i=1}^d (1-\phi_q(x^i)) \leq \frac{2}{q}\sum_{i=1}^d |\phi_q'(x^i)| \leq \frac{2\sqrt{d}}{q}\left\|\nabla \overline{H}_d(x)\right\| \leq \left\|\nabla \overline{H}_d(x)\right\|,$$

where the second inequality follows by (77), the third inequality follows by Cauchy-Schwarz, and the last uses $2\sqrt{d}/q = 1/(8\sqrt{d}) \leq 1$ for $d \geq 2$.

We obtain that in both cases, the inequality $\overline{H}_d(x) \leq \left\|\nabla \overline{H}_d(x)\right\|$ is satisfied, which concludes the proof. □

Lemma 26 ($\overline{H}_d$ is a zero-chain)**.** *Let $\overline{H}_d$ be defined as in* (80)*. Then, $\overline{H}_d$ is a zero-chain with threshold $\tau = 1/2q$.*

Proof. Let $\tau \coloneqq \mathrm{prog}_{1/2q}(x)$. Our goal is to show $\frac{\partial}{\partial x^i}\overline{H}_d(x) = 0$ for all $i > \tau + 1$. From Lemma 21, we have that $\psi_q(t) = 0$ and $\psi_q'(t) = 0$ for $t \leq 1/2q$.

Let us consider $i > \tau + 1$, then the indices $i-1$ and i both exceed τ. This implies that $|x^{i-1}| \leq 1/2q$ and $|x^i| \leq 1/2q$, giving $\psi_q(x^{i-1}) = 0$ and $\psi_q'(x^i) = 0$. For $i < d$ this yields

$$\frac{\partial}{\partial x^i}\overline{H}_d(x) = -\psi_q(x^{i-1})\phi_q'(x^i) - \psi_q'(x^i)\phi_q(x^{i+1}) = 0,$$

and for $i = d$ (when $d > \tau + 1$),

$$\frac{\partial}{\partial x^d}\overline{H}_d(x) = -\psi_q(x^{d-1})\phi_q'(x^d) = 0.$$

Hence $\mathrm{supp}(\nabla \overline{H}_d(x)) \subseteq \{1, \ldots, \tau+1\}$. □

Lemma 27 (Smooth KŁ chain)**.** *Let $d \geq 2$, $L > 0$, $q \coloneqq 16d$. Let ψ_q, ϕ_q be defined as in* (72)*, and set $x^0 = 1$. We define $H_d(x)$ as follows*

$$H_d(x) \coloneqq \frac{L}{454q^2}\left(d - \psi_q(x^0)\phi_q(x^1) - \sum_{i=2}^d \psi_q(x^{i-1})\phi_q(x^i)\right), \quad x \in \mathbb{R}^d. \tag{81}$$

Then H_d satisfies the following properties.

(i) H_d *is* L*-smooth on* $\mathbb{R}^d$*;*

(ii) It has the unique global minimizer $x^\star = (1, \dots, 1)$ *with* $H_d(x^\star) = 0$*;*

(iii) On the box $[0, 2]^d$*, it satisfies* μ*-KŁ condition with* $\mu = 1$

$$\|\nabla H_d(x)\| \geq H_d(x) - H_d(x^\star).$$

(iv) H_d *is a zero-chain with threshold* $\tau := 1/2q$*;*

(v) For all $x \in \mathbb{R}^d$ *such that* $\mathrm{prog}_\tau(x) < d$*, and* $j = \mathrm{prog}_\tau(x) + 1$ *we have*

$$\left|\frac{\partial}{\partial x^j} H_d(x)\right| \leq \frac{L}{3632d}; \tag{82}$$

(vi) Let $s \in \{0, 1, \dots, d\}$ *and assume that* $\mathrm{supp}(x) \subseteq \{1, \dots, s\}$*. Then,* $\overline{H}_d(x) \geq d - s$ *and*

$$H_d(x) - H_d(x^\star) = H_d(x) \geq \frac{L}{454q^2}(d - s). \tag{83}$$

Proof. Note that $H_d = \frac{L}{454q^2}\overline{H}_d$ with $\overline{H}_d$ as in (80).

Smoothness. By Lemma 23, $\left\|\nabla^2 \overline{H}_d(x)\right\|_{\mathrm{op}} \leq 454\, q^2$ for all $x \in \mathbb{R}^d$. Hence

$$\left\|\nabla^2 H_d(x)\right\|_{\mathrm{op}} = \frac{L}{454q^2}\left\|\nabla^2 \overline{H}_d(x)\right\|_{\mathrm{op}} \leq \frac{L}{454q^2} \cdot 454q^2 = L.$$

Global minimizer. By Lemma 24, $x^\star = (1, \dots, 1) \in \mathbb{R}^d$ is the unique global minimizer of $\overline{H}_d$ with $\overline{H}_d(x^\star) = 0$. Since $\frac{L}{454q^2} > 0$, the same holds for H_d.

KŁ inequality. By Lemma 25, $\left\|\nabla \overline{H}_d(x)\right\| \geq \overline{H}_d(x) - \overline{H}_d(x^\star)$ for all $x \in [0, 2]^d$. Multiplying by $\frac{L}{454q^2} > 0$ gives,

$$\|\nabla H_d(x)\| \geq H_d(x) - H_d(x^\star) \quad \text{for all } x \in [0, 2]^d.$$

Zero-chain. By Lemma 26, $\mathrm{supp}(\nabla \overline{H}_d(x)) \subseteq \{1, \dots, \mathrm{prog}_\tau(x) + 1\}$ for all $x \in \mathbb{R}^d$. Since $\nabla H_d = \frac{L}{454q^2}\nabla \overline{H}_d$, the supports coincide and the same inclusion holds for H_d.

Frontier coordinate upper bound. Let $m = \mathrm{prog}_\tau(x) < d$ and $j = m+1$, then $|x^j| \leq \tau = 1/(2q)$ by definition of $\mathrm{prog}_\tau(x)$. Therefore, we have $x^j \leq \tau$ and hence $\psi_q'(x^j) = 0$ by Lemma 21. The coordinate derivative of the unscaled chain $\overline{H}_d(x)$ is

$$\frac{\partial}{\partial x^i}\overline{H}_d(x) = -\psi_q(x^{i-1})\phi_q'(x^i) - \psi_q'(x^i)\phi_q(x^{i+1}), \quad \text{for } 1 \leq i < d,$$

and

$$\frac{\partial}{\partial x^d}\overline{H}_d(x) = -\psi_q(x^{d-1})\phi_q'(x^d).$$

Therefore, for the frontier coordinate j,

$$\frac{\partial}{\partial x^j}\overline{H}_d(x) = -\psi_q(x^{j-1})\phi_q'(x^j),$$

with the same formula when $j = d$. Since $0 \leq \psi_q \leq 1$ and $|\phi_q'| \leq 2q$ by Lemmas 21 and 22,

$$\left|\frac{\partial}{\partial x^j} H_d(x)\right| = \frac{L}{454q^2}\left|\frac{\partial}{\partial x^j}\overline{H}_d(x)\right| \leq \frac{L}{454q^2} \cdot 1 \cdot 2q = \frac{2L}{454q} = \frac{L}{3632d}.$$

Suffix suboptimality. The assumption means that $x^{s+1} = \cdots = x^d = 0$. It implies that

$$\psi_q(x^s)\phi_q(x^{s+1}) = \psi_q(x^s)\phi_q(0) = 0.$$

For every $i > s+1$, one has $x^{i-1} = 0$, and hence

$$\psi_q(x^{i-1}) = \psi_q(0) = 0.$$

Thus all summands with index $i > s$ vanish. The remaining s summands satisfy $|\psi_q(x^{i-1})\phi_q(x^i)| \leq 1$, since $0 \leq \psi_q \leq 1$ and $|\phi_q| \leq 1$ by (73) of Lemma 21 and (75) of Lemma 22. Therefore, we have

$$\sum_{i=1}^{d} \psi_q(x^{i-1})\phi_q(x^i) \leq s,$$

and hence for the function $\overline{H}_d(x)$, defined in (80)

$$\overline{H}_d(x) = d - \sum_{i=1}^{d} \psi_q(x^{i-1})\phi_q(x^i) \geq d - s.$$

Multiplying by $\frac{L}{454q^2}$ gives the statement for $H_d(x)$. □

G.3 The Unbiased Stochastic Masking Oracle

In this section, we construct the stochastic oracle g that realizes H_d as a stochastic zero-chain in the sense of Definition G.2.

Definition G.3 (Stochastic masking oracle)**.** *Let $\sigma \geq 0$ be the noise variance scale. Let $b_d \coloneqq L/3632d$ be the frontier coordinate upper bound from* (82) *of Lemma 27. Let*

$$\pi \coloneqq \frac{b_d^2}{b_d^2 + \sigma^2} \in (0, 1],$$

and define a random variable $\zeta \sim \mathrm{Bernoulli}(\pi)$, which sampled independently at each query. With $m(x) \coloneqq \mathrm{prog}_\tau(x)$ and $j(x) \coloneqq m(x) + 1$, define the stochastic gradient

$$g(x) \coloneqq \begin{cases} \nabla H_d(x), & x \notin [0,2]^d \text{ or } m(x) = d, \\ \nabla H_d(x) + \left(\dfrac{\zeta}{\pi} - 1\right) \frac{\partial}{\partial x^{j(x)}} H_d(x) e_{j(x)}, & x \in [0,2]^d \text{ and } m(x) < d, \end{cases}$$

where e_i is the i^{th} standard basis unit vector.

In the noisy case every coordinate is returned exactly except the frontier coordinate $j(x)$, which is rescaled by $1/\pi$ and revealed only when $\zeta = 1$.

Lemma 28 (Unbiasedness and variance bound)**.** *Let g satisfy Definition G.3. Then, for every $x \in \mathbb{R}^d$,*

$$\mathbb{E}[g(x) \mid x] = \nabla H_d(x).$$

Moreover, for every $x \in [0,2]^d$,

$$\mathbb{E}\left[\|g(x) - \nabla H_d(x)\|^2 \mid x\right] \leq \sigma^2.$$

Proof. If $x \notin [0,2]^d$ or if $m(x) = d$, the oracle is exact and the claims are immediate. Thus, we assume that $x \in [0,2]^d$ and $m(x) < d$. For shortness, we define $j = j(x)$. Since $\mathbb{E}[\zeta] = \pi$,

$$\begin{aligned} \mathbb{E}[g(x) \mid x] &= \mathbb{E}[\nabla H_d(x)] + \mathbb{E}\left[\left(\frac{\zeta}{\pi} - 1\right) \frac{\partial}{\partial x^j} H_d(x) e_j\right] \\ &= \nabla H_d(x) + \left(\frac{\mathbb{E}[\zeta]}{\pi} - 1\right) \frac{\partial}{\partial x^j} H_d(x) e_j \\ &= \nabla H_d(x). \end{aligned}$$

Thus, the oracle is unbiased. Now we need to bound the variance. Note that only coordinate j is random. Hence

$$g(x) - \nabla H_d(x) = \left(\frac{\zeta}{\pi} - 1\right) \frac{\partial}{\partial x^j} H_d(x) e_j,$$

and therefore

$$\mathbb{E}\left[\|g(x) - \nabla H_d(x)\|^2 \mid x\right] = \left|\frac{\partial}{\partial x^j} H_d(x)\right|^2 \mathbb{E}\left[\left(\frac{\zeta}{\pi} - 1\right)^2\right].$$

For $\zeta \sim \text{Bernoulli}(\pi)$, we have

$$\mathbb{E}\left[\left(\frac{\zeta}{\pi} - 1\right)^2\right] = \frac{1-\pi}{\pi}.$$

By (82) of Lemma 27, we have $\left|\frac{\partial}{\partial x^j} H_d(x)\right| \le b_d$. Hence

$$\mathbb{E}\left[\|g(x) - \nabla H_d(x)\|^2 \mid x\right] \le b_d^2 \frac{1-\pi}{\pi}.$$

By the definition of π,

$$\frac{1-\pi}{\pi} = \frac{\sigma^2}{b_d^2}.$$

Consequently,

$$\mathbb{E}\left[\|g(x) - \nabla H_d(x)\|^2 \mid x\right] \le \sigma^2.$$

For $x \notin [0,2]^d$, the variance is zero by definition, so the bound $\mathbb{E}\left[\|g(x) - \nabla H_d(x)\|^2 \mid x\right] \le \sigma^2$ holds for all $x \in \mathbb{R}^d$. □

G.4 Zero-respecting Stochastic Methods

In this section, we formalize the class of zero-respecting stochastic methods, together with the natural filtration generated by their interaction with the oracle.

Definition G.4 (Zero-respecting stochastic method)**.** *Let a stochastic first-order method be initialized at $x_0 = (0, 0, \ldots, 0) \in \mathbb{R}^d$. Let $x_1, x_2, \ldots \in [0,2]^d$ be the iterates of the method, while $g(x_k)$ be received stochastic gradient oracles. The method updates x_{k+1} such that it is measurable with respect to $\mathcal{F}_{k+1}$. It is* zero-respecting *if, almost surely,*

$$\operatorname{supp}(x_{k+1}) \subseteq \bigcup_{s=0}^{k} \operatorname{supp}(g(x_s)) \quad \textit{for all } k \ge 0,$$

and the output $\widehat{x}_k \in [0,2]^d$ after K calls satisfies the same condition with the union over $s = 0, \ldots, K-1$.

Definition G.5 (Natural filtration)**.** *For $k \ge 0$, set*

$$\mathcal{F}_k \coloneqq \sigma\left(x_0, g(x_0), x_1, g(x_1), \ldots, x_{k-1}, g(x_{k-1}), x_k\right).$$

Thus, x_k is $\mathcal{F}_k$-measurable and $g(x_k)$ is sampled conditionally on $\mathcal{F}_k$, and x_{k+1} is measurable with respect to $\mathcal{F}_{k+1} = \sigma(\mathcal{F}_k, g(x_k), x_{k+1})$. We define the largest revealed index before step k,

$$A_k \coloneqq \max\left(\{0\} \cup \bigcup_{s=0}^{k-1} \operatorname{supp}(g(x_s))\right), \quad A_0 \coloneqq 0.$$

Since $g(x_0), \ldots, g(x_{k-1})$ are all $\mathcal{F}_k$-measurable, so is A_k. At step k, the oracle draws $\zeta_k \sim \text{Bernoulli}(\pi)$ independently of $\mathcal{F}_k$ when $\text{prog}_\tau(x_k) < d$, and sets $\zeta_k \coloneqq 0$ otherwise. Define $S_K \coloneqq \sum_{k=0}^{K-1} \zeta_k$.

Lemma 29 (Reveal-time domination)**.** *For every $K \ge 1$, we have $A_K \le S_K$ almost surely and $\mathbb{E}[S_K] \le \pi K$. Moreover, $\text{prog}_\tau(\widehat{x}_k) \le A_K \le S_K$ almost surely.*

Proof. Because the method is zero-respecting, $\text{prog}_\tau(x_k) \le A_k$. Let $m_k \coloneqq \text{prog}_\tau(x_k)$. By the zero-chain property,

$$\operatorname{supp}(\nabla H_d(x_k)) \subseteq \{1, \ldots, m_k + 1\}.$$

The stochastic oracle masks the frontier coordinate $m_k + 1$. If $\zeta_k = 0$, then the returned stochastic gradient $g(x_k)$ has no nonzero coordinate beyond m_k, and hence no nonzero coordinate beyond A_k.

If $\zeta_k = 1$, then the returned gradient stochastic gradient may contain coordinate $m_k + 1$. Since $m_k \le A_k$, this can increase the largest revealed coordinate by at most one. Therefore

$$A_{k+1} \le A_k + \zeta_k.$$

Since $A_0 = 0$, induction gives

$$A_K \le \sum_{k=0}^{K-1} \zeta_k = S_K.$$

Since x_k is $\mathcal{F}_k$-measurable and ζ_k is drawn independently of $\mathcal{F}_k$,

$$\mathbb{E}[\zeta_k \mid \mathcal{F}_k] = \pi \cdot \mathbf{1}\left[\mathrm{prog}_\tau(x_k) < d\right] \le \pi.$$

By the tower property,

$$\mathbb{E}[S_K] = \sum_{k=0}^{K-1} \mathbb{E}\left[\mathbb{E}[\zeta_k \mid \mathcal{F}_k]\right] \le \pi K.$$

Finally, the output is zero-respecting, so

$$\mathrm{prog}_\tau(\widehat{x}_k) \le A_K.$$

Combining this with $A_K \le S_K$ proves the claim. □

G.5 Lower Bound for Function Suboptimality

We now replace the stationarity target by a function-gap target. The target accuracy is denoted by $\varepsilon > 0$:

$$\mathbb{E}[H_d(x) - H_d(x^\star)] \le \varepsilon.$$

The proof uses the suffix suboptimality (83) of Lemma 27. The intuition behind the proof is that before many coordinates have been revealed, a long unrevealed suffix still contributes a large amount of function gap.

Theorem 11 (Expected suboptimality lower bound)**.** *Let $0 < \varepsilon \le L/1859584$, $d := \lfloor L/(464896\varepsilon) \rfloor$, $q := 16d$. Let H_d be the chain defined in Lemma 27 and g be the stochastic gradient oracle satisfying Definition G.3 with masking probability π. Every zero-respecting stochastic method with iterates in $[0,2]^d$ and output $\widehat{x}_K$ after K iterations satisfies*

$$\mathbb{E}\left[H_d(\widehat{x}_K) - H_d(x^\star)\right] \ge \frac{L}{454q^2}(d - \pi K)_+,$$

where $(r)_+ := \max\{r, 0\}$. Consequently, any method guaranteeing $\mathbb{E}[H_d(\widehat{x}_K) - H_d(x^\star)] \le \varepsilon$ must satisfy

$$K = \Omega\left(\frac{L}{\varepsilon} + \frac{L\sigma^2}{\varepsilon^3}\right).$$

Proof. By the zero-respecting property of the method of the output,

$$\mathrm{supp}(\widehat{x}_K) \subseteq \{1, \dots, A_K\}.$$

Then, from (83) of Lemma 27, we obtain

$$H_d(\widehat{x}_K) - H_d(x^\star) \ge \frac{L}{454q^2}(d - A_K).$$

Since $A_K \le \min\{S_K, d\}$, then we get

$$d - A_K \ge (d - S_K)_+.$$

Therefore

$$\mathbb{E}\left[H_d(\widehat{x}_K) - H_d(x^\star)\right] \ge \frac{L}{454q^2}\mathbb{E}[(d - S_K)_+].$$

The function $u \mapsto (d-u)_+$ is convex, so Jensen's inequality gives

$$\mathbb{E}[(d - S_K)_+] \ge (d - \mathbb{E}[S_K])_+.$$

Using Lemma 29, $\mathbb{E}[S_K] \le \pi K$, hence

$$\mathbb{E}\left[H_d(\widehat{x}_K) - H_d(x^\star)\right] \ge \frac{L}{454q^2}(d - \pi K)_+.$$

If $K \le d/(2\pi)$, then

$$\frac{L}{454q^2}(d - \pi K)_+ \ge \frac{Ld}{908q^2}.$$

The condition on ε implies that $L/(464896\varepsilon) \ge 4$. Then, by the definition of d, we have $d \ge 4$. Also,

$$d \le \frac{L}{464896\varepsilon}.$$

Therefore

$$\frac{L}{454q^2}\frac{d}{2} = \frac{L}{454(16d)^2}\frac{d}{2} = \frac{L}{232448d} \ge 2\varepsilon.$$

Thus, any method with expected gap at most ε must have $K > \frac{d}{2\pi}$. It remains to evaluate $\frac{d}{2\pi}$. Since

$$\frac{1}{\pi} = 1 + \frac{\sigma^2}{b_d^2},$$

we get

$$\frac{d}{2\pi} = \frac{d}{2} + \frac{d\sigma^2}{2b_d^2}.$$

Using $b_d = \frac{L}{3632d}$,

$$\frac{d\sigma^2}{2b_d^2} = \frac{d\sigma^2}{2} \cdot \frac{3632^2 d^2}{L^2} = \frac{6595712\sigma^2 d^3}{L^2}.$$

Because $\frac{L}{464896\varepsilon} \ge 4$, the floor satisfies

$$d = \left\lfloor \frac{L}{464896\varepsilon} \right\rfloor \ge \frac{L}{929792\varepsilon}.$$

Consequently,

$$\frac{d}{2} \ge \frac{L}{1859584\varepsilon},$$

and

$$\frac{6595712\sigma^2 d^3}{L^2} \ge \frac{6595712\sigma^2}{L^2}\left(\frac{L}{929792\varepsilon}\right)^3 = \frac{L\sigma^2}{2^{29} \cdot 227\varepsilon^3}.$$

Therefore, this implies that

$$K \ge \frac{d}{2\pi} \ge \frac{L}{1859584\varepsilon} + \frac{L\sigma^2}{2^{29} \cdot 227\varepsilon^3},$$

as stated in the theorem. □

G.6 Detailed Comparison of Lower bounds in this Work and Masiha et al. (2026a)

In stochastic optimization under gradient domination, Fontaine et al. (2021); Fatkhullin et al. (2022) analyze first-order methods and Masiha et al. (2022) second-order methods; here we compare our lower bound with that of Masiha et al. (2026a).

Both our lower bound and that of Masiha et al. (2026a) use the probabilistic zero-chain framework introduced in Arjevani et al. (2023). Therefore they share the common underlying mechanism. However, the explicit hard functions the two constructions use are different. Specifically in Masiha et al. (2026a) the base chain in their construction is the following:

$$F_N(x) := \varphi(x_0) + \sum_{i=0}^{N-2} \Theta(x_i)\varphi(x_{i+1}) + \sum_{i=0}^{N-1} B(x_i), \qquad x \in \mathbb{R}^N. \tag{84}$$

$$\Theta(t) \coloneqq \begin{cases} 0, & t \le \frac{1}{2}, \\ 3(2t-1)^2 - 2(2t-1)^3, & \frac{1}{2} < t < 1, \\ 1, & t \ge 1, \end{cases} \tag{85}$$

$$\varphi(t) \coloneqq 1 - \exp\left(-\frac{(t-1)^2}{2}\right), \tag{86}$$

$$B(t) \coloneqq \begin{cases} 2 + \frac{1}{2}\left(t + \frac{1}{2}\right)^2, & t \le -\frac{1}{2}, \\ 2, & -\frac{1}{2} < t \le \frac{1}{2}, \\ 2\big(1 - \Theta(t)\big), & \frac{1}{2} < t < 1, \\ \frac{1}{2}(t-1)^2, & t \ge 1. \end{cases} \tag{87}$$

Here, Θ acts as an activation gate, φ measures the error associated with the next coordinate, and (B) is a separable coordinate-wise potential. In particular, $B(0) = 2$ and $B'(0) = 0$, so each hidden coordinate contributes a positive amount to the objective without directly producing a gradient signal. While our hard function does not have the summation of B functions, specifically a complete description of our hard function can be found in Section G.2, but we restate it below for completeness:

$$\overline{H}_d(x) \coloneqq d - \psi_q(x^0)\phi_q(x^1) - \sum_{i=2}^{d} \psi_q(x^{i-1})\phi_q(x^i) \quad \text{for all} \quad x \in \mathbb{R}^d, \tag{88}$$

where we have:

$$\chi(s) \coloneqq \begin{cases} e^{-1/s} & s > 0 \\ 0 & s \le 0 \end{cases}, \qquad \omega(s) \coloneqq \frac{\chi(s)}{\chi(s) + \chi(1-s)}, \tag{89}$$

$$\psi_q(t) \coloneqq \omega(2qt - 1), \qquad \phi_q(t) \coloneqq \frac{1 - (1-t)^q}{1 + (1-t)^q}. \tag{90}$$

The functions Θ, ψ_q that links the loss to the next coordinate each time are different, as well as the loss functions φ, ϕ_q. Additionally our hard instance does not have the term with the summation over the losses B.

These differentiations lead to an additional structural difference, while both functions are both KŁ and smooth at the distance where the iterates land, our function remains KŁ and not PŁ arbitrarily close to the minimizer *while the construction in Masiha et al. (2026a) becomes PŁ close to the minimizer*. This local PŁ behavior does not invalidate their lower bound: during the relevant oracle budget, some hidden coordinates remain below the activation threshold with controlled probability, and hence the algorithm has not yet entered this neighborhood of the minimizer.