BEYOND QUADRATIC LOSS: THE STABILITY PHASE DIAGRAM OF ADAM

Anonymous authors
Paper under double-blind review

ABSTRACT

Loss spikes are recurrent instabilities in neural-network training and can arise from multiple mechanisms. For Adam in particular, macroscopic loss spikes have been linked to optimizer dynamics, yet how its two momentum timescales govern them remains unclear. We investigate this dependence by mapping training dynamics across the (β_1, β_2) plane. Across a range of model–task settings, an approximately linear boundary, $1-\beta_2 = C(1-\beta_1)$, separates spiky from non-spiky dynamics, whereas a one-dimensional quadratic loss produces approximately cubic slope. A one-dimensional superquadratic loss $L(x) \propto |x|^n$ recovers the near-linear scaling and links the boundary coefficient to the effective loss exponent n. We further show that confident cross-entropy losses develop a core–wall landscape comprising a narrow quadratic core followed by a steep wall, which produces effective superquadratic behavior at the scale of an optimizer update. Together, these results connect Adam loss spikes to both the mismatch between momentum timescales and finite-scale superquadratic loss geometry beyond the Hessian.

1 INTRODUCTION

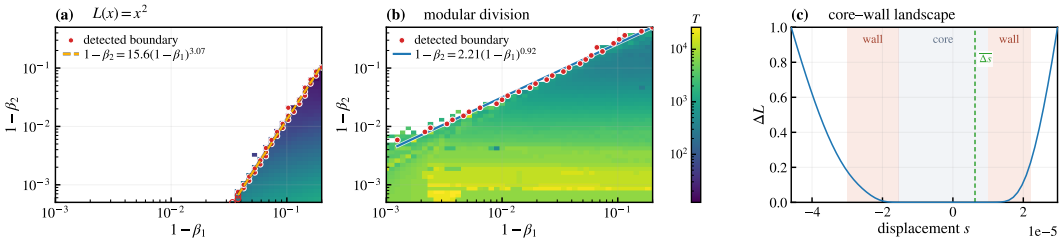

Figure 1: Overview of the empirical boundary scaling and finite-scale loss geometry. (a) Adam stability phase diagram in the (β_1, β_2) plane for $L(x) = x^2/2$. A power-law fit gives $1-\beta_2 \simeq 15.6(1-\beta_1)^{3.07}$, showing the approximately cubic scaling of the quadratic model. (b) Phase diagram for the modular-division Transformer, whose detected boundary follows the near-linear relation $1-\beta_2 \simeq 2.21(1-\beta_1)^{0.92}$. In (a,b), color denotes the measured oscillation period. (c) Directional loss slice $L(s) = L(\theta + s\hat{d}_t)$ at a pre-spike point of the modular-division Transformer, where $\hat{d}_t$ is the Adam-preconditioned gradient direction. The green dashed line shows the magnitude of a typical update, which is comparable to the flat-core width.

Loss spikes are abrupt, macroscopic loss excursions that recur across neural-network training settings. Their causes depend on the training regime. In Adam training, such spikes have been widely reported Chowdhery et al. (2023); Molybog et al. (2023); Thilak et al. (2022). Under a local quadratic approximation, gradient descent is stable when $\lambda_{\max}(H_t) < 2/\eta$. Crossing this threshold can induce Edge-of-Stability dynamics characterized by non-monotonic loss excursions and spikes Xing et al. (2018); Jastrzebski et al. (2020); Cohen et al. (2021). For adaptive optimizers, the analogous criterion replaces the raw Hessian with a preconditioned Hessian, leading to the Adaptive Edge of Stability for Adam and RMSProp Cohen et al. (2023; 2025). Sustained violations of this effective stability threshold have also been linked to macroscopic Adam loss spikes Bai et al.

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.

(2026b). Other mechanisms apply in more specific settings. Numerical Feature Inflation explains spikes in low-precision models trained with cross-entropy loss Liu et al. (2026), while weight-norm criticality applies to models with scale-invariant layers Li et al. (2026).

Adam's first and second gradient moments have memory timescales set by β_1 and β_2, respectively Kingma & Ba (2015). **We investigate how the mismatch between β_1 and β_2 governs loss spikes.** We focus on macroscopic loss spikes triggered by violations of Adam's EoS condition, rather than the microscopic oscillations associated with short-period attractors Bock & Weiß (2019); Fong & Yang (2026). Across diverse models and tasks, we identify the near-linear stability boundary $1-\beta_2 \propto (1-\beta_1)$ shown in Figure 1(b). At fixed β_1, the spiky phase lies on the larger-β_2 side of this boundary. To isolate the underlying dynamics, we study Adam on the one-dimensional quadratic loss $L(x) = kx^2/2$. This model instead produces the approximately cubic boundary $1-\beta_2 \propto (1-\beta_1)^3$ in Figure 1(a). Bai et al. (2026a) analyze the (β_1, β_2) phase transition on even-degree degenerate polynomials, obtaining a linear boundary but a phase ordering different from that observed in real models. These discrepancies motivate an analysis beyond the conventional infinitesimal quadratic approximation.

We then analyze the Adam stability phases for a superquadratic one-dimensional loss $L(x) = |x|^n$ with $n > 2$. We find a wedge-shaped spiky phase bounded approximately by

$$C_L(1-\beta_1) \geq 1-\beta_2 \geq C_R(1-\beta_1). \tag{1}$$

The right boundary follows from the competition between the effective learning rate η_{eff} and the critical learning rate η_{crit}, giving $C_R = 2(n-2)/n$. The left boundary involves more complex oscillatory dynamics and is described empirically by $C_L \simeq 12(n-2)/(3n-4)$. The right boundary is consistent with Bai et al. (2026a), but does not appear in real models because the second derivative at the minimum is nonzero. The boundary we observe in real models instead corresponds to the left boundary.

Local loss landscapes are often analyzed in the infinitesimal limit through a second-order Taylor expansion. In neural networks, the corresponding Hessian spectra typically contain a broad near-zero bulk and a few isolated outliers Sagun et al. (2018). This structure has motivated river-valley descriptions of neural-network optimization Xing et al. (2018); Wen et al. (2024).

We instead analyze directional loss landscapes along the preconditioned gradient at finite scales and find a **core–wall landscape** composed of a flat quadratic core, a steep superquadratic wall, and an outer rollover, as illustrated in Figure 1(c). **We show that this structure arises naturally from confident cross-entropy and derive the scale of its quadratic core.** During training, the landscape undergoes finite-scale progressive sharpening, in which the preconditioned sharpness increases while the flat core contracts. Once a typical update becomes comparable to the core width, Adam probes the wall and encounters a large effective exponent n. We estimate this exponent at pre-spike points and substitute it into the empirical relation for C_L. The resulting estimates capture the scale of the observed boundary coefficients.

This work makes three contributions. First, we identify a near-linear phase boundary separating spiky and non-spiky Adam dynamics across diverse models and tasks. Second, we study Adam stability in the context of superquadratic losses and show how the memory timescales of the first and second moments set the phase boundaries. Third, we study the local loss landscape at finite scales, show that its core–wall structure arises naturally from confident cross-entropy, and relate its effective exponent at the optimizer's update scale to the empirical boundary coefficient.

2 Stability Phase Boundaries Across Six Models

We study six model–task settings: one-layer Transformers Vaswani et al. (2017) for modular division and addition modulo 53, a character-level Transformer for next-character prediction on Tiny Shakespeare, an MLP and a VGG11-style CNN Simonyan & Zisserman (2015) for CIFAR-10 classification Krizhevsky (2009), and a convolutional autoencoder for binarized MNIST reconstruction LeCun et al. (1998). All models use AdamW Loshchilov & Hutter (2019) with cross-entropy for classification and language modeling or binary cross-entropy for reconstruction. Within each setting, only β_1 and β_2 vary, while the architecture, data, random seed, and other optimizer hyperparameters remain fixed. Appendix B.1 gives the full configurations.

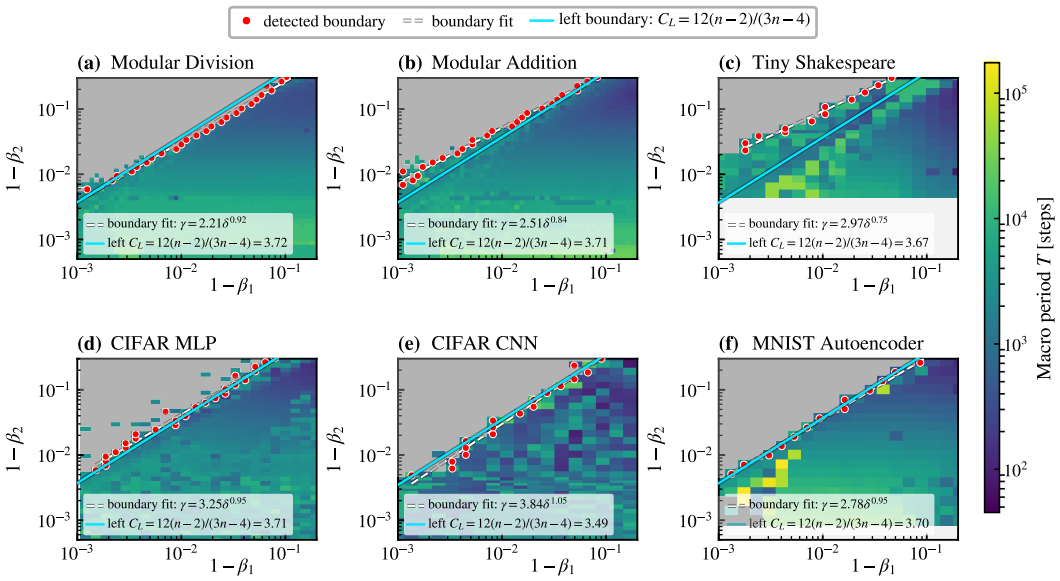

Figure 2: Phase boundaries for six models. (a–f) Modular division, modular addition, Tiny Shakespeare, a CIFAR-10 MLP, a CIFAR-10 CNN, and an MNIST autoencoder, respectively. The axes are the memory gaps $\delta = 1 - \beta_1$ and $\gamma = 1 - \beta_2$, and color indicates the detected macroscopic oscillation period in optimizer steps. Red points mark boundary locations, white dashed curves show power-law fits $\gamma = C\delta^p$, and solid cyan lines show estimates from the effective loss exponents in Table 1 and the empirical relation $C_L = 12(n-2)/(3n-4)$ in Section 3. Gray cells have no accepted period. Unsampled regions are uncolored.

We scan logarithmically spaced memory gaps $\delta = 1 - \beta_1$ and $\gamma = 1 - \beta_2$. All six diagrams cover $\delta \in [10^{-3}, 0.2]$ and $\gamma \in [5 \times 10^{-4}, 0.3]$. After the initial transient, we extend trajectories with unresolved periods and identify macroscopic oscillations from low-frequency spectral peaks that meet the prominence criterion in Appendix B.2.

For each fixed γ, the boundary is the immediate right neighbor of the largest sampled δ with no accepted period. We fit these points over the selected ranges using $\log\gamma = \log C + p\log\delta$ to estimate C and p. **All six phase diagrams approach the linear scaling** $\gamma \propto \delta$, with fitted exponents $p = 0.92, 0.84, 0.75, 0.95, 1.05, 0.95$ for (a–f), respectively. The region above the boundary, where $(1-\beta_2)/(1-\beta_1)$ is larger, generally contains no detected macroscopic spikes. Within the spiky region, periods tend to increase as β_2 approaches one and depend more weakly on β_1.

3 One-dimensional toy model of Adam stability

To isolate the boundary mechanism, we first state the relevant local stability condition.

Lemma 1 (Local stability condition for Adam)**.** *Consider a network in a locally quadratic region with Hessian H_t and preconditioner $D_t = \mathrm{diag}[(\sqrt{\hat{v}_t} + \epsilon)^{-1}]$. With both held fixed locally and late-time bias corrections neglected, Adam is unstable if the following condition holds Cohen et al. (2023); Bai et al. (2026b):*

$$\frac{1-\beta_1}{1+\beta_1}\lambda_{\max}(D_t H_t) > \frac{2}{\eta}. \tag{2}$$

Appendix A.1 gives the derivation. Its one-dimensional specialization follows immediately.

Corollary 2 (Stability condition for Adam in 1D)**.** *For one-dimensional curvature $\lambda_t = L''(x_t) > 0$, define*

$$\eta_{\mathrm{eff,t}} = \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon}, \qquad \eta_{\mathrm{crit,t}} = \frac{2(1+\beta_1)}{(1-\beta_1)\lambda_t}. \tag{3}$$

Under the same local approximation, Adam is unstable when $\eta_{\mathrm{eff,t}} > \eta_{\mathrm{crit,t}}$.

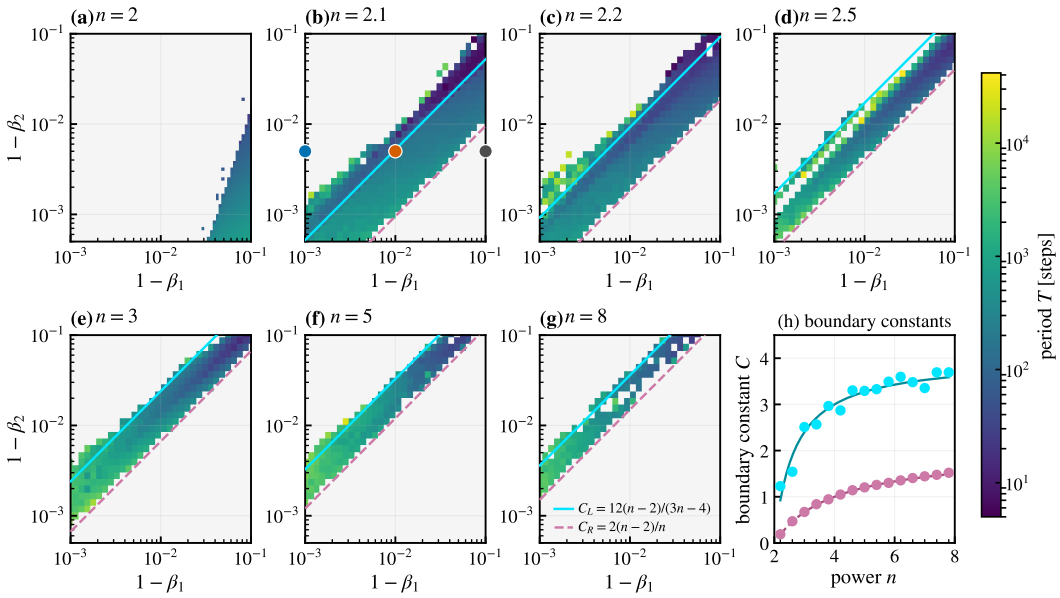

Figure 3: Phase diagrams for the one-dimensional losses $L(x) = k|x|^n/n$ with $k = 1$. (a–g) Results for $n = 2, 2.1, 2.2, 2.5, 3, 5, 8$, respectively, where $\delta = 1 - \beta_1$, $\gamma = 1 - \beta_2$, and color indicates the period. Solid cyan lines show the empirical left boundary $\gamma = 12(n-2)\delta/(3n-4)$, and dashed pink lines show the predicted right boundary $\gamma = 2(n-2)\delta/n$. Both coefficients vanish at $n = 2$, where the boundary is approximately cubic. The blue, orange, and gray points in (b) identify the trajectories in Figure 4. (h) Measured boundary coefficients as functions of n.

We first study the quadratic loss $L(x) = kx^2/2$ using Adam with $k = 1$, learning rate $\eta = 0.1$, numerical stabilizer $\epsilon = 10^{-30}$, no weight decay, $x_0 = 1$, and $m_0 = v_0 = 0$. Each beta pair is trained for 250,000 updates, of which the first 125,000 are discarded. We estimate the period from upward crossings of the local stability threshold and set $T = 125{,}000/N_{\text{cross}}$ when at least three crossings are detected. Figure 3(a) shows the approximately cubic boundary $\gamma \simeq 15.6\delta^{3.07}$, consistent with Figure 1(a). Repeated crossings occur mainly below this boundary. Appendix C.2 examines the effect of weight decay on 1-dimensional models.

The boundary remains approximately cubic across the tested k and η, and also across ϵ while $\epsilon \ll \sqrt{\hat{v}_t}$. Appendix C.1 reports these controls, which support an explanation based on the relative memory timescales of the two moments. Because the quadratic exponent differs substantially from the near-unit neural-network exponents, we next consider $L(x) = k|x|^n/n$. For $1 < n < 2$, divergent curvature near zero produces rapid oscillations rather than separated macroscopic spikes. We therefore focus on the superquadratic case $n > 2$.

For $n > 2$, Figure 3(b–g) shows a wedge-shaped spiky region between two lines of approximately unit log–log slope,

$$C_L(1 - \beta_1) \geq 1 - \beta_2 \geq C_R(1 - \beta_1). \tag{4}$$

We derive the right boundary from Adam's local stability condition and characterize the left boundary empirically. The boundaries observed in the six neural-network phase diagrams correspond to the left boundary C_L. The right boundary C_R requires the loss to remain superquadratic asymptotically as the displacement approaches zero. As Section 4 shows, the measured neural-network landscapes instead enter a quadratic core in this limit, so the right-boundary mechanism does not apply to them.

During a quiet interval, $(1 - \beta_2)g_{t+1}^2 \ll \beta_2 v_t$ makes self-decay dominate the second moment, giving $v_t \propto \beta_2^t$ (Liu, 2026). Lemma 8 makes this decay asymptotically exact for bounded rescaled trajectories. While $\sqrt{\hat{v}_t} \gg \epsilon$, this decay raises $\eta_{\text{eff,t}}$. The approach to zero on a superquadratic loss simultaneously lowers the curvature and raises $\eta_{\text{crit,t}}$. Figure 4(d–f) illustrates three resulting regimes. If $\eta_{\text{crit,t}}$ grows faster, $\eta_{\text{eff,t}}$ cannot catch it and sustained macroscopic spikes are suppressed,

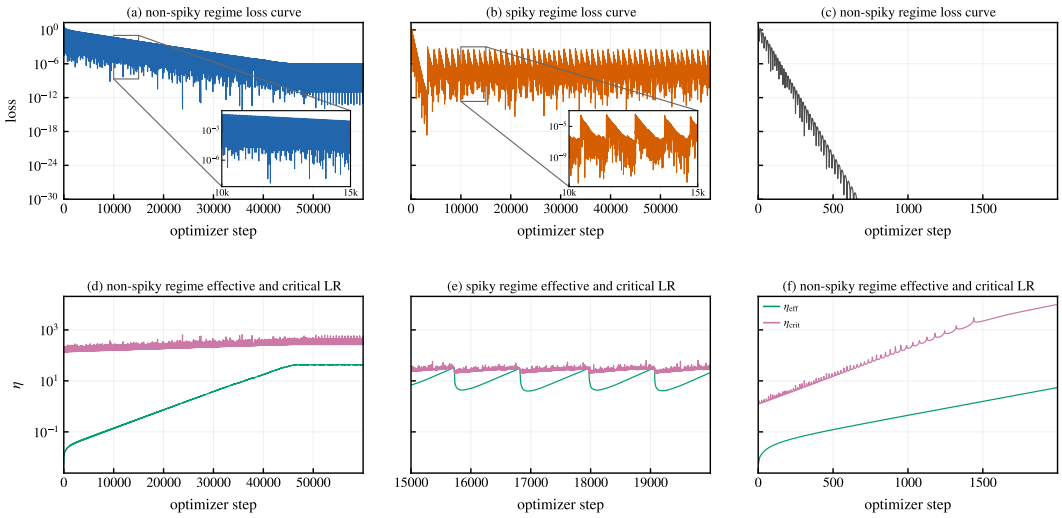

Figure 4: Loss and learning-rate dynamics for $n = 2.1$. The columns correspond to $(\beta_1, \beta_2) = (0.999, 0.995)$, $(0.99, 0.995)$, and $(0.9, 0.995)$, matching the blue, orange, and gray points in Figure 3(b). (a–c) Loss curves. (d–f) Effective learning rate η_{eff} in green and critical rate η_{crit} in pink. In (d), rapid oscillations persist while η_{eff} remains below η_{crit} and eventually saturates. In (e), η_{eff} repeatedly reaches the threshold, producing spikes that reduce it. In (f), faster growth of η_{crit} suppresses sustained macroscopic spikes.

as in panel (f). If $\eta_{\text{eff,t}}$ repeatedly reaches the threshold, each crossing triggers a spike whose large gradients replenish v_t and lower $\eta_{\text{eff,t}}$, as in panel (e). Alternatively, the trajectory can be captured by a short-period microscopic attractor, where $\eta_{\text{eff,t}}$ saturates below $\eta_{\text{crit,t}}$ and rapid oscillations persist without macroscopic spikes, as in panel (d). Theorem 3 formalizes the first regime for trajectories attracted to the rescaled fixed points. We absorb $1/n$ into k and write $L(x) = k|x|^n$, which leaves the boundary coefficients unchanged.

Theorem 3 (Right boundary for $L(x) = k|x|^n$ models)**.** *Consider Adam applied to the loss landscape $L(x) = k|x|^n$, where $\frac{1}{2} < \beta_1 < 1$, $0 < \beta_2 < 1$, and $n > 2$. In the late-training regime, neglect ϵ and the bias-correction factors, then trajectories eventually cease to exhibit EoS-triggered spikes when*

$$\beta_1 \beta_2^{-n/[2(n-2)]} < 1. \tag{5}$$

Proof sketch. An equivalent condition was derived by Bai et al. (2026a) from the local stability of the full normalized Adam dynamics on even-degree polynomials. Lemma 8 shows that the full update is a summable perturbation of the limiting recurrence below for every real $n > 2$.

$$y_{t+1} = a y_t - b y_{t-1} - c|y_t|^{n-2} y_t, \qquad b = \beta_1 \beta_2^{-n/[2(n-2)]}, \tag{6}$$

where a, b, and c are time independent. Whenever the nonzero fixed points exist within the theorem's parameter range, both satisfy $S_* < 1$. Linearization and the Jury conditions show that these fixed points are stable exactly when $b < 1$, so attracted trajectories eventually cease to spike. The boundary $b = 1$ gives $\beta_2 = \beta_1^{2(n-2)/n}$. Appendix A.2 provides the fixed-point bounds and full stability calculation. □

Along these trajectories, $x_t \sim q^t y_*$ gives $g_t^2 / v_t \propto q^{2t} \to 0$, consistent with the self-decay approximation. In the phase diagram coordinates, the right boundary is

$$\gamma_R(\delta, n) = 1 - (1 - \delta)^{2(n-2)/n} = \frac{2(n-2)}{n} \delta + O(\delta^2). \tag{7}$$

The plotted line uses the leading small-δ coefficient $C_R = 2(n-2)/n$. The $n = 2$ boundary is not linear and instead exhibits the approximately cubic scaling in Figure 3(a).

These regimes show that the left boundary involves more complex oscillatory dynamics and finite-step effects, and a complete analytical explanation remains open. We summarize its dependence on n by the empirical relation $C_L \simeq 12(n-2)/(3n-4)$, which captures the trend in Figure 3(h).

4 CORE–WALL LANDSCAPE AND ITS EXPONENTIAL-SUM MECHANISM

The one-dimensional results suggest that the near-linear β-boundary reflects a superquadratic loss profile at the scale explored by Adam. Although the conventional quadratic approximation remains valid sufficiently close to a low-loss minimum, its domain of validity can be much smaller than an optimizer update Ma et al. (2022a). The optimizer then probes the steep finite-scale growth outside this neighborhood, motivating our **core–wall landscape** description. It refines the usual river–valley picture by resolving the basin around a low-loss minimum into an inner quadratic core, a superquadratic wall, and an outer rollover. Related notions of flatness have been connected to generalization Hochreiter & Schmidhuber (1997); Keskar et al. (2017), while basin-like parameter regions have been used to study behavioral preservation and fine-tuning robustness Peng et al. (2024); Chen et al. (2026). **We quantify this directional geometry using an effective logarithmic exponent**. Experimentally, we observe the core–wall landscape at pre-spike points in all six model–task settings. Appendix D.1 details the pre-spike selection procedure and illustrates how contraction of the quadratic core toward the optimizer's update scale produces progressive sharpening at finite scales.

Definition 4 (Core–wall landscape)**.** For a frozen direction $\hat{d}$, define the directional slice $\phi(s) = L(\theta + s\hat{d})$ and let s_* be one of its local minima. For $r > 0$, define the two excess-loss branches and their local exponents by

$$\Delta L_{\pm}(r) = \phi(s_* \pm r) - \phi(s_*), \qquad n_{\mathrm{eff},\pm}(r) = \frac{\mathrm{d}\log \Delta L_{\pm}(r)}{\mathrm{d}\log r}. \tag{8}$$

As r increases, the branches pass successively through a quadratic core with $n_{\mathrm{eff},\pm} \simeq 2$, a superquadratic wall beginning at the first crossing of the operational threshold $n_{\mathrm{eff},\pm} = 2.5$, and an outer rollover in which $n_{\mathrm{eff},\pm}$ decreases from its wall value. We call this ordered basin structure the *core–wall landscape*.

To quantify this observation across the six tasks in Figure 2, we evaluate low-loss checkpoints along the Adam-preconditioned gradient direction

$$\phi_t(s) = L(\theta_t + s\hat{d}_t), \qquad \hat{d}_t = \frac{D_t g_t}{\|D_t^{1/2} g_t\|_2}. \tag{9}$$

During descent and near the bottoms of temporal loss valleys, these slices exhibit a core–wall landscape. Conventional progressive sharpening tracks the growth of Hessian or preconditioned-Hessian sharpness Cohen et al. (2021; 2023). Our slices reveal a finite-scale counterpart in which the quadratic core also contracts toward the optimizer's update scale, as illustrated in Appendix Figure 9. We next show that this geometry arises naturally from confident cross-entropy losses. **The flat core follows from Hessian collapse as the predicted probabilities approach their hard targets.**

Lemma 5 (Hessian collapse for confident cross-entropy)**.** *On a fixed dataset of N samples with one-hot targets y_i, let $L(\theta) = N^{-1}\sum_i \ell_i(\theta)$ be the unregularized softmax cross-entropy, with twice differentiable logits $z_i(\theta)$ and probabilities $p_i = \mathrm{softmax}(z_i)$. Along any parameter sequence such that $p_i \to y_i$ for every sample, assume that $J_i = \partial z_i/\partial\theta$ and $\nabla_\theta^2 z_{ik}$ remain uniformly bounded. Then $\|\nabla_\theta^2 L\|_2 \to 0$.*

Proof sketch. The generalized Gauss–Newton decomposition writes each sample Hessian as $J_i^\top C_i J_i + \sum_k (p_{ik} - y_{ik})\nabla_\theta^2 z_{ik}$, where $C_i = \mathrm{diag}(p_i) - p_i p_i^\top$. If $\varepsilon_i = 1 - p_{i,c_i}$, then $\|C_i\|_2 \le 2\varepsilon_i$ and $\|p_i - y_i\|_1 = 2\varepsilon_i$. The assumed derivative bounds therefore make both terms $O(\varepsilon_i)$, and the finite average vanishes as every $\varepsilon_i \to 0$. Appendix A.3 gives the full norm bounds. □

Lemma 5 identifies the origin of the flat core but does not determine the size of the neighborhood in which the quadratic approximation is accurate. We derive this scale from the exponential-sum structure of confident cross-entropy.

Definition 6 (Confident cross-entropy loss slice)**.** Let $i \in \{1, \dots, N\}$ index samples or tokens, let y_i be the correct class for sample i, and let $j \neq y_i$ index its incorrect classes. Along the frozen slice $\theta(s) = \theta_* + s\hat{d}$, define the correct-versus-incorrect logit margin as $m_{ij}(s) = z_{i,y_i}(s) - z_{ij}(s)$. The exact cross-entropy slice is

$$L_{\mathrm{CE}}(s) = \frac{1}{N} \sum_{i=1}^{N} \log \left(1 + \sum_{j \neq y_i} e^{-m_{ij}(s)} \right). \tag{10}$$

In the confident regime, the relevant margins are large and positive, so $\log(1+u) \simeq u$. If they are locally linearized as $m_{ij}(s) \simeq m_{ij} - a_{ij}s$, where $m_{ij} = m_{ij}(0)$ and $a_{ij} = -\mathrm{d}m_{ij}(s)/\mathrm{d}s|_{s=0}$, we call

$$L_{\mathrm{conf}}(s) = \frac{1}{N} \sum_{i} \sum_{j \neq y_i} e^{-m_{ij} + a_{ij}s}. \tag{11}$$

the *confident cross-entropy loss slice*.

In confident cross-entropy, the large margins $m \gg 1$ suppress the baseline loss, $L_{\mathrm{conf}}(0) \propto \sum e^{-m} \simeq 0$, whereas the directional slopes a can have much larger magnitudes. At the local loss minimum in Figure 5(a), for example, $\langle m \rangle_w \pm \mathrm{std}_w(m) = 17.08 \pm 1.29$, whereas $\langle a \rangle_w \simeq 1.33 \times 10^2$ and $\sigma_a \simeq 2.30 \times 10^5$. Such large slope magnitudes make factors of the form e^{-m+as} grow exponentially under very small displacements, **leaving only an $O(1/\sigma_a)$ neighborhood in which the quadratic approximation remains accurate.** The following theorem makes this scale precise under a Gaussian model for a.

Theorem 7 (Quadratic-core scale of confident cross-entropy)**.** *Approximate the CE-weighted distribution of directional margin slopes by a Gaussian, $a \overset{w}{\sim} \mathcal{N}(\mu_a, \sigma_a^2)$. Within this Gaussian model, with the origin chosen at the minimum of the confident surrogate and $\sigma_a > 0$, the slice has a quadratic-core scale $s_{\mathrm{core}} = O(1/\sigma_a)$.*

Proof sketch. The normalized confident loss is the moment-generating function of the weighted slope distribution. Gaussianity and stationarity at $s = 0$ give

$$\frac{\Delta L_{\mathrm{conf}}(s)}{L_{\mathrm{conf}}(0)} = e^{(\sigma_a s)^2/2} - 1, \qquad n_{\mathrm{eff}} = \frac{(\sigma_a s)^2 e^{(\sigma_a s)^2/2}}{e^{(\sigma_a s)^2/2} - 1}. \tag{12}$$

Thus $n_{\mathrm{eff}} = 2 + (\sigma_a s)^2/2 + O((\sigma_a s)^4)$, so departure from the quadratic core occurs at $|s| = O(1/\sigma_a)$; the threshold $n_{\mathrm{eff}} = 2.5$ gives $|s| \simeq 0.964/\sigma_a$. Appendix A.4 contains the full derivation. □

To resolve the local shape, we refine s_* for each frozen slice and plot $\Delta L = \phi_t(s) - \phi_t(s_*)$ against $|s - s_*|$ on logarithmic axes. Figure 5 compares a measured modular-division slice with a synthetic confident cross-entropy example. We construct the latter as $L_{\mathrm{CE}}(s) = \log(1 + \sum_{i=1}^{16} e^{-M_i + a_i s})$, with $M_i \sim \mathrm{Uniform}[10, 20]$ and $a_i \sim \mathcal{N}(0, 3000^2)$, then shift the slopes so that $\sum_i e^{-M_i} a_i = 0$ and $s_* = 0$. In both cases, the logarithmic slope rises from an approximately quadratic core to a superquadratic wall and then decreases in the outer rollover.

For the measured slice in Figure 5(a), averaging the two branches gives $n_{\mathrm{eff}} \simeq 2.18$ in the core and $n_{\mathrm{eff}} \simeq 10.90$ along the wall. The core–wall transition occurs at $O(10^{-6})$, comparable to the typical update scale $\overline{\Delta s} \simeq 6.3 \times 10^{-6}$, while the rollover begins at $O(10^{-5})$. Consistently, $1/\sigma_a \simeq 4.35 \times 10^{-6}$ for the measured slice. The synthetic slice in Figure 5(b) exhibits the same three regimes with $1/\sigma_a \simeq 2.97 \times 10^{-4}$.

Beyond the core, exponential reweighting favors terms with increasingly extreme directional slopes. If $\langle a \rangle_s$ denotes their loss-weighted mean at displacement s, then

$$\frac{\mathrm{d} \log L_{\mathrm{conf}}}{\mathrm{d} \log |s|} = s \langle a \rangle_s. \tag{13}$$

The growth of $|\langle a \rangle_s|$ produces a superquadratic wall that admits a finite-interval power-law approximation, yielding the effective exponent used in the one-dimensional model. At larger displacements,

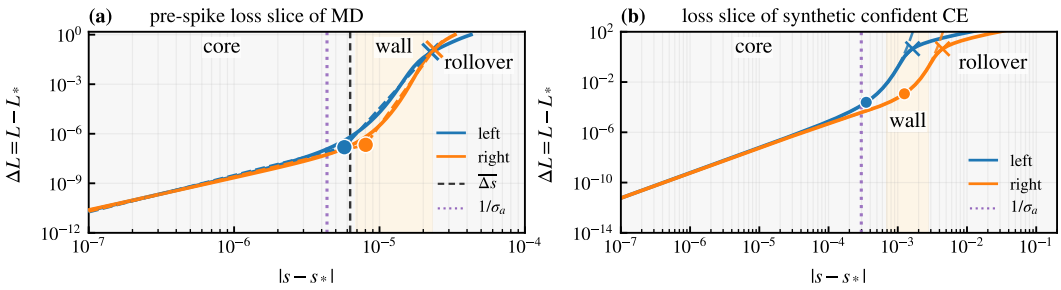

Figure 5: Directional loss slices shown as $\Delta L = L - L_*$ versus $|s - s_*|$ on log–log axes. (a) The selected modular-division pre-spike point at step 37,210 for $(\beta_1, \beta_2) = (0.9, 0.999)$. Solid curves are the measured left and right branches, dashed curves are piecewise core–wall landscape fits, circles and crosses mark the fitted knots, and the vertical dashed line marks the typical projected update $\overline{\Delta s}$. Shading identifies the core, wall, and rollover fit intervals. (b) A synthetic confident cross-entropy slice. Solid curves show the exact loss, and dashed curves show its confident approximation. Purple dotted lines mark $1/\sigma_a$. In both examples, the wall emerges at the exponential-sum scale.

Table 1: Effective wall exponents and the resulting empirical left-boundary coefficients for the six model–task settings.

Model–task setting	n_{wall}	Estimated C_L
Modular division	10.90	3.72
Modular addition	10.58	3.71
Tiny Shakespeare	9.41	3.67
CIFAR-10 MLP	10.48	3.71
CIFAR-10 CNN	6.52	3.49
MNIST autoencoder	10.22	3.70

the logarithm in exact cross-entropy changes a dominant contribution from exponential to asymptotically linear growth, reducing the effective exponent and producing the rollover.

Finally, we connect these directional measurements to the β-phase diagrams. We use the fitted wall exponent over the resolved intermediate regime as an effective input to the empirical left-boundary relation from Section 3,

$$C_L(n_{\text{wall}}) \simeq \frac{12(n_{\text{wall}} - 2)}{3n_{\text{wall}} - 4}, \qquad 1 - \beta_2 = C_L(n_{\text{wall}})(1 - \beta_1). \tag{14}$$

The non-spiky region below the right boundary of the pure power-law model arises asymptotically as $s \to 0$ while the loss remains superquadratic. In the measured neural-network landscapes, however, this limit enters the quadratic core, cutting off the pure-superquadratic right-boundary mechanism before it can be observed. We therefore compare the measured wall exponent only with the empirical left boundary.

For a controlled comparison, all six model–task settings use the same six reference beta pairs and apply the same core–wall–rollover fitting procedure at the pre-spike points. We first obtain one jointly fitted n_{wall} for each beta pair and then report the unweighted mean over the six pairs in Table 1. We evaluate C_L from this mean exponent. Appendix D.1 describes how the pre-spike points are located, and Appendix D.2 gives the directional-slice fit and aggregation procedure.

5 Conclusion and Outlook

Taken together, our results make three contributions. First, they establish a near-linear phase boundary between spiky and non-spiky Adam dynamics across diverse models and tasks. Second, our analysis attributes the shift from cubic to linear boundary scaling to superquadratic loss geometry and shows how the first- and second-moment memory timescales set the phase boundaries. Third, we

study local loss landscapes at finite scales and show that confident cross-entropy produces a core–wall structure consisting of a narrow quadratic core, a superquadratic wall, and an outer rollover. The effective exponent measured at the optimizer's update scale predicts the empirical boundary coefficient, directly linking finite-scale landscape geometry to the beta phase diagram.

Looking ahead, three questions appear especially promising.

1. Basin- and valley-like loss landscapes arise in broader underparameterized neural-network settings Bosman et al. (2020); Ruiz-Garcia et al. (2021). Can the core–wall exponential-sum mechanism also explain these landscapes?
2. More generally, non-quadratic geometry with an exponent n_{eff} that changes across direction, scale, and training time is likely to be the norm. Can an optimizer estimate n_{eff} at its update scale and adapt its learning rate or moment timescales to the evolving phase boundary?
3. What mechanism drives progressive sharpening at finite scales? How do the margins m_{ij} and their directional derivatives $a_{ij} = -\mathrm{d}m_{ij}/\mathrm{d}s$ evolve during training?

6 Related work

Edge of Stability (EoS). Trajectory studies linked learning rates to sharp directions and showed that progressive sharpening reaches the quadratic threshold $2/\eta$, producing EoS dynamics Xing et al. (2018); Jastrzębski et al. (2018); Jastrzebski et al. (2020); Cohen et al. (2021). Adaptive EoS uses the preconditioned Hessian, central-flow theory describes averaged oscillations, and decoupling between g_t^2 and v_t can trigger Adam spikes Cohen et al. (2023; 2025); Bai et al. (2026b). Low-dimensional analyses cover quadratic two-cycles and frozen-EoS restoration Bock & Weiß (2019); Fong & Yang (2026). Bai et al. (2026a) analyze the (β_1, β_2) phase transition on even-degree degenerate polynomials, whereas we study it on the core–wall landscapes observed in real models. Owing to the quadratic core, at fixed β_1 the spiky phase in real models occurs at larger β_2, in direct contrast to their prediction. Complementary work studies near-zero Hessian bulk, loss slices, and river-valley dynamics Sagun et al. (2018); Li et al. (2018); Wen et al. (2024).

Loss spikes. Loss spikes in large-language-model training may require checkpoint-and-data interventions Chowdhery et al. (2023), while unusually large Adam updates can become weakly aligned with descent Molybog et al. (2023). Momentum-dependent oscillations, spikes, and divergence have also been observed Ma et al. (2022b). Proposed mechanisms include lower-loss-as-sharper geometry and lagging second moments that amplify preconditioned curvature even on quadratics Li et al. (2023); Bai et al. (2026b). Slingshot connects cyclic adaptive-optimizer instability to grokking, although neither implies the other Power et al. (2022); Thilak et al. (2022). Numerical Feature Inflation and weight-norm criticality explain spikes in low precision or scale-invariant networks with weight decay Liu et al. (2026); Li et al. (2026). Our one-dimensional examples attribute beta-boundary scaling to competing moment timescales and superquadratic finite-scale geometry.

Core–wall landscapes. Flat minima have been associated with generalization Hochreiter & Schmidhuber (1997); Keskar et al. (2017), although parameter-space sharpness is not invariant to function-preserving reparameterizations Dinh et al. (2017). Volume flatness, worst-case loss increases, and Hessian sharpness address different questions. Recent LLM studies define basins by preserving alignment or task performance under parameter perturbations Peng et al. (2024); Chen et al. (2026). These characterize fine-tuning robustness, whereas our core–wall landscape uses local training loss along a dynamically selected direction to explain spike onset.

Optimization with non-quadratic curvature. Beyond the quadratic threshold, prior work studies unstable convergence, multiscale subquadratic landscapes, and cubic self-stabilization Ahn et al. (2022); Ma et al. (2022a); Damian et al. (2023). Generalized self-concordance bounds finite-displacement departures for logistic loss, while the generalized Gauss–Newton decomposition shows why pointwise curvature need not determine finite-step dynamics Bach (2010); Schraudolph (2002). Neither derives the directional scale $O(1/\sigma_a)$ of a confident cross-entropy exponential sum.

AI USE STATEMENT

Generative AI tools were used to assist with proof development, method implementation and software development, data analysis, visualization, and manuscript formatting. The authors reviewed all AI-assisted work and take responsibility for the final content of this work, including all claims and artifacts produced with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

Appendix A provides the derivations and proofs for the theoretical results. Appendix B reports the model architectures, datasets, optimizer settings, training horizons, random seeds, and period-detection procedure for the six full-model experiments. Appendix C gives the parameter-sensitivity and weight-decay controls for the one-dimensional models. Appendix D describes the pre-spike selection, directional-slice construction, wall-exponent fitting, bootstrap uncertainty estimation, and aggregation procedure used in the finite-scale landscape analysis.

REFERENCES

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 247–257, Baltimore, Maryland, USA, 2022. PMLR. URL https://proceedings.mlr.press/v162/ahn22a.html.

Francis Bach. Self-concordant analysis for logistic regression. *Electronic Journal of Statistics*, 4:384–414, 2010. doi: 10.1214/09-EJS521. URL https://doi.org/10.1214/09-EJS521.

Zhiwei Bai, Jiajie Zhao, Zhangchen Zhou, Zhi-Qin John Xu, and Yaoyu Zhang. Towards understanding Adam convergence on highly degenerate polynomials, 2026a. URL https://arxiv.org/abs/2603.09581. Accepted at ICML 2026.

Zhiwei Bai, Zhangchen Zhou, Jiajie Zhao, Xiaolong Li, Zhiyu Li, Feiyu Xiong, Hongkang Yang, Yaoyu Zhang, and Zhi-Qin John Xu. Adaptive preconditioners trigger loss spikes in Adam, 2026b. URL https://arxiv.org/abs/2506.04805. Accepted at ICML 2026.

Sebastian Bock and Martin Georg Weiß. Non-convergence and limit cycles in the Adam optimizer. In Igor V. Tetko, Věra Kůrková, Pavel Karpov, and Fabian Theis (eds.), *Artificial Neural Networks and Machine Learning–ICANN 2019: Deep Learning*, volume 11728 of *Lecture Notes in Computer Science*, pp. 232–243, Cham, 2019. Springer International Publishing. doi: 10.1007/978-3-030-30484-3_20. URL https://doi.org/10.1007/978-3-030-30484-3_20.

Anna Sergeevna Bosman, Andries P. Engelbrecht, and Mardé Helbig. Visualising basins of attraction for the cross-entropy and the squared error neural network loss functions. *Neurocomputing*, 400: 113–136, 2020. doi: 10.1016/j.neucom.2020.02.113. URL https://doi.org/10.1016/j.neucom.2020.02.113.

Tony Cai, Jianqing Fan, and Tiefeng Jiang. Distributions of angles in random packing on spheres. *Journal of Machine Learning Research*, 14(57):1837–1864, 2013. URL https://jmlr.org/papers/v14/cai13a.html.

Huanran Chen, Yinpeng Dong, Zeming Wei, Yao Huang, Yichi Zhang, Hang Su, and Jun Zhu. Unveiling the basin-like loss landscape in large language models, 2026. URL https://arxiv.org/abs/2505.17646. Published at ICLR 2026.

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*, 24(240):1–113, 2023. URL https://www.jmlr.org/papers/v24/22-1144.html.

Jeremy M. Cohen, Simran Kaur, Yuanzhi Li, J. Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *Proceedings of the Ninth International Conference on Learning Representations*, Virtual Conference, 2021. OpenReview.net. URL https://openreview.net/forum?id=jh-rTtvkGeM.

Jeremy M. Cohen, Behrooz Ghorbani, Shankar Krishnan, Naman Agarwal, Sourabh Medapati, Michal Badura, Daniel Suo, David Cardoze, Zachary Nado, George E. Dahl, and Justin Gilmer.

Adaptive gradient methods at the edge of stability, 2023. URL https://arxiv.org/abs/2207.14484. NeurIPS 2023 Workshop on Heavy Tails in Machine Learning.

Jeremy M. Cohen, Alex Damian, Ameet Talwalkar, J. Zico Kolter, and Jason D. Lee. Understanding optimization in deep learning with central flows, 2025. URL https://arxiv.org/abs/2410.24206. Published at ICLR 2025.

Alex Damian, Eshaan Nichani, and Jason D. Lee. Self-stabilization: The implicit bias of gradient descent at the edge of stability. In *Proceedings of the Eleventh International Conference on Learning Representations*, Kigali, Rwanda, 2023. OpenReview.net. URL https://openreview.net/forum?id=nhKHA59gXz.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028, Sydney, Australia, 2017. PMLR. URL https://proceedings.mlr.press/v70/dinh17b.html.

Yiman Fong and Heng Yang. Provable edge-of-stability for Adam on a one-dimensional quadratic, 2026. URL https://arxiv.org/abs/2608.20638.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997. doi: 10.1162/neco.1997.9.1.1. URL https://doi.org/10.1162/neco.1997.9.1.1.

Stanisław Jastrzębski, Zachary Kenton, Nicolas Ballas, Asja Fischer, Yoshua Bengio, and Amos Storkey. On the relation between the sharpest directions of DNN loss and the SGD step length, 2018. URL https://arxiv.org/abs/1807.05031.

Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The break-even point on optimization trajectories of deep neural networks, 2020. URL https://arxiv.org/abs/2002.09572.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima, 2017. URL https://arxiv.org/abs/1609.04836. Published at ICLR 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the Third International Conference on Learning Representations*, San Diego, California, USA, 2015. International Conference on Learning Representations. doi: 10.48550/arXiv.1412.6980. URL https://arxiv.org/abs/1412.6980.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Ontario, Canada, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791. URL https://doi.org/10.1109/5.726791.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets, 2018. URL https://arxiv.org/abs/1712.09913. Advances in Neural Information Processing Systems 31.

Xiaolong Li, Zhi-Qin John Xu, and Zhongwang Zhang. Loss spike in training neural networks, 2023. URL https://arxiv.org/abs/2305.12133.

Xiaolong Li, Zhangchen Zhou, and Zhi-Qin John Xu. Weight-norm criticality: A mechanism for loss spikes induced by the normalization and weight decay, 2026. URL https://arxiv.org/abs/2607.21005.

Hanqing Liu, Jianjun Cao, Yuanze Li, and Zijian Zhou. Grokking or glitching? how low-precision drives slingshot loss spikes, 2026. URL https://arxiv.org/abs/2605.06152.

Ziming Liu. Optimization 4 – loss spikes, January 2026. URL https://KindXiaoming.github.io/blog/2026/optimization-4/.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the Seventh International Conference on Learning Representations*, New Orleans, Louisiana, USA, 2019. OpenReview.net. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Chao Ma, Daniel Kunin, Lei Wu, and Lexing Ying. Beyond the quadratic approximation: The multiscale structure of neural network loss landscapes. *Journal of Machine Learning*, 1(3): 247–267, 2022a. doi: 10.4208/jml.220404. URL https://www.global-sci.com/jml/article/view/13179.

Chao Ma, Lei Wu, and Weinan E. A qualitative study of the dynamic behavior for adaptive gradient algorithms. In Joan Bruna, Jan Hesthaven, and Lenka Zdeborova (eds.), *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference*, volume 145 of *Proceedings of Machine Learning Research*, pp. 671–692, Virtual Conference, 2022b. PMLR. URL https://proceedings.mlr.press/v145/ma22a.html.

Igor Molybog, Peter Albert, Moya Chen, Zachary DeVito, David Esiobu, Naman Goyal, Punit Singh Koura, Sharan Narang, Andrew Poulton, Ruan Silva, Binh Tang, Diana Liskovich, Puxin Xu, Yuchen Zhang, Melanie Kambadur, Stephen Roller, and Susan Zhang. A theory on Adam instability in large-scale machine learning, 2023. URL https://arxiv.org/abs/2304.09871.

ShengYun Peng, Pin-Yu Chen, Matthew Hull, and Duen Horng Chau. Navigating the safety landscape: Measuring risks in finetuning large language models, 2024. URL https://arxiv.org/abs/2405.17374.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets, 2022. URL https://arxiv.org/abs/2201.02177.

Miguel Ruiz-Garcia, Ge Zhang, Samuel S. Schoenholz, and Andrea J. Liu. Tilting the playing field: Dynamical loss functions for machine learning. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9157–9167, Virtual, 2021. PMLR. URL https://proceedings.mlr.press/v139/ruiz-garcia21a.html.

Levent Sagun, Utku Evci, V. Ugur Güney, Yann Dauphin, and Leon Bottou. Empirical analysis of the hessian of over-parametrized neural networks, 2018. URL https://arxiv.org/abs/1706.04454. ICLR 2018 Workshop Track.

Nicol N. Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002. doi: 10.1162/08997660260028683. URL https://doi.org/10.1162/08997660260028683.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the Third International Conference on Learning Representations*, San Diego, California, USA, 2015. International Conference on Learning Representations. doi: 10.48550/arXiv.1409.1556. URL https://arxiv.org/abs/1409.1556.

Vimal Thilak, Etai Littwin, Shuangfei Zhai, Omid Saremi, Roni Paiss, and Joshua Susskind. The slingshot mechanism: An empirical study of adaptive optimizers and the grokking phenomenon, 2022. URL https://arxiv.org/abs/2206.04817.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, Long Beach, California, USA, 2017. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Kaiyue Wen, Zhiyuan Li, Jason Wang, David Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape perspective, 2024. URL https://arxiv.org/abs/2410.05192.

Chen Xing, Devansh Arpit, Christos Tsirigotis, and Yoshua Bengio. A walk with SGD, 2018. URL https://arxiv.org/abs/1802.08770.

A PROOFS

A.1 PROOF OF LEMMA 1

Proof. For gradient descent, the local perturbation vector evolves as

$$\delta_{t+1} = (I - \eta H_t)\delta_t. \tag{15}$$

Along an eigenvector of H_t with eigenvalue λ, this becomes

$$\delta_{t+1} = (1 - \eta\lambda)\delta_t. \tag{16}$$

Stability requires $|1 - \eta\lambda| < 1$, which gives

$$\eta\lambda < 2 \qquad \Longrightarrow \qquad \lambda < \frac{2}{\eta}. \tag{17}$$

With first-moment momentum,

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)g_t, \qquad \theta_{t+1} = \theta_t - \eta m_t. \tag{18}$$

Linearizing $g_t \approx H_t\delta_t$ and eliminating m_t gives the vector recurrence

$$\delta_{t+1} = \left[(1 + \beta_1)I - \eta(1 - \beta_1)H_t\right]\delta_t - \beta_1\delta_{t-1}. \tag{19}$$

Along a Hessian eigen-direction, the characteristic equation is

$$r^2 - \alpha r + \beta_1 = 0, \qquad \alpha = (1 + \beta_1) - \eta(1 - \beta_1)\lambda. \tag{20}$$

The Jury stability condition gives

$$\eta(1 - \beta_1)\lambda < 2(1 + \beta_1), \tag{21}$$

or equivalently

$$\frac{1 - \beta_1}{1 + \beta_1}\lambda < \frac{2}{\eta}. \tag{22}$$

Adam replaces the raw gradient by a coordinate-wise normalized gradient:

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)g_t^2, \qquad \theta_{t+1} = \theta_t - \eta\frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}. \tag{23}$$

Because v_t tracks squared gradients, its RMS scale is $\sqrt{\hat{v}_t}$, giving the preconditioner

$$D_t = \operatorname{diag}\left(\frac{1}{\sqrt{\hat{v}_t} + \epsilon}\right). \tag{24}$$

Freezing D_t and H_t replaces the local operator H_t by D_tH_t. In the absence of first-moment momentum, the corresponding perturbation dynamics are

$$\delta_{t+1} \approx (I - \eta D_t H_t)\delta_t. \tag{25}$$

The matrix D_tH_t is similar to the symmetric matrix $D_t^{1/2}H_tD_t^{1/2}$, so its eigenvalues are real. Including first-moment momentum and applying the preceding condition to the largest eigenvalue yields the instability criterion

$$\frac{1 - \beta_1}{1 + \beta_1}\lambda_{\max}(D_tH_t) > \frac{2}{\eta}, \tag{26}$$

which is the Adam EoS instability condition. □

A.2 Proof of Theorem 3

Lemma 8 (Second-moment self-decay). *Let $g_t = kn|x_t|^{n-2}x_t$, $v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2$, and $v_0 > 0$. Set $q = \beta_2^{1/[2(n-2)]}$, $x_t = q^t y_t$, and $V_t = \beta_2^{-t} v_t$. If $\sup_t |y_t| < \infty$, then V_t converges to a finite limit $V_\infty > 0$, and*

$$v_t = V_\infty \beta_2^t [1 + O(q^{2t})], \qquad \frac{g_t^2}{v_t} = O(q^{2t}), \qquad \frac{v_t}{\beta_2 v_{t-1}} = 1 + O(q^{2t}). \tag{27}$$

Proof. The rescaled second moment obeys the exact recurrence

$$V_t - V_{t-1} = (1-\beta_2)\beta_2^{-t} g_t^2 = (1-\beta_2)(kn)^2 q^{2t} |y_t|^{2(n-1)}. \tag{28}$$

If $|y_t| \le M$, the increments are nonnegative and bounded by a summable geometric sequence. Hence

$$V_\infty = v_0 + (1-\beta_2)(kn)^2 \sum_{s=1}^{\infty} q^{2s} |y_s|^{2(n-1)} \tag{29}$$

exists, is finite, and is strictly positive. Moreover,

$$0 \le V_\infty - V_t \le \frac{(1-\beta_2)(kn)^2 M^{2(n-1)}}{1-q^2} q^{2(t+1)}. \tag{30}$$

Thus $V_t = V_\infty[1 + O(q^{2t})]$, which proves the first claim after multiplying by β_2^t. The remaining two claims follow from

$$\frac{g_t^2}{v_t} = \frac{(kn)^2 q^{2t} |y_t|^{2(n-1)}}{V_t}, \qquad \frac{v_t}{\beta_2 v_{t-1}} = \frac{V_t}{V_{t-1}}, \tag{31}$$

because $V_t \ge v_0 > 0$ and $V_t - V_{t-1} = O(q^{2t})$. □

Proof of Theorem 3. Using the notation of Lemma 8, a trajectory covered by the theorem satisfies $\eta_{\mathrm{eff,t}} = \eta/\sqrt{v_t}$, $V_t \to V_\infty > 0$, and $V_\infty - V_t = O(q^{2t})$. Define the stability ratio

$$S_t = \frac{\eta_{\mathrm{eff,t}}}{\eta_{\mathrm{crit,t}}} = \frac{(1-\beta_1)kn(n-1)\eta}{2(1+\beta_1)\sqrt{V_t}} |y_t|^{n-2}. \tag{32}$$

A trajectory that eventually remains at $S_t < 1$ has no further threshold crossings and hence no further spikes triggered by this mechanism.

Eliminating the first moment from the exact discrete updates gives

$$\begin{aligned} x_{t+1} = &\left(1 + \beta_1 \sqrt{\frac{v_{t-1}}{v_t}}\right) x_t - \beta_1 \sqrt{\frac{v_{t-1}}{v_t}} x_{t-1} \\ &- (1-\beta_1) kn \frac{\eta}{\sqrt{v_t}} |x_t|^{n-2} x_t. \end{aligned} \tag{33}$$

Under this rescaling, the dynamics become the asymptotically autonomous recurrence

$$y_{t+1} = a_t y_t - b_t y_{t-1} - c_t |y_t|^{n-2} y_t, \tag{34}$$

where

$$R_t = \sqrt{\frac{V_{t-1}}{V_t}}, \qquad a_t = \frac{1 + \beta_1 \beta_2^{-1/2} R_t}{q}, \qquad b_t = \frac{\beta_1 R_t}{q^n}, \qquad c_t = \frac{(1-\beta_1)kn\eta}{q\sqrt{V_t}}. \tag{35}$$

The lemma gives $R_t = 1 + O(q^{2t})$, so these coefficients converge at a summable rate to

$$a = \frac{1 + \beta_1 \beta_2^{-1/2}}{q}, \qquad b = \frac{\beta_1}{q^n} = \beta_1 \beta_2^{-n/[2(n-2)]}, \qquad c = \frac{(1-\beta_1)kn\eta}{q\sqrt{V_\infty}}. \tag{36}$$

The two nonzero fixed points exist when $a - 1 - b > 0$, equivalently $bq < 1$, and satisfy

$$y_* = \pm \left(\frac{a-1-b}{c}\right)^{1/(n-2)}. \tag{37}$$

Both have the same stability ratio because S depends only on $|y|$:

$$S_* = \frac{(n-1)q(a-1-b)}{2(1+\beta_1)} = \frac{(n-1)(1-q)(1-bq)}{2(1+\beta_1)}. \tag{38}$$

Existence implies $q > \beta_1^{1/(n-1)}$, while $q < 1$ implies $bq > \beta_1$. Therefore,

$$S_* < \frac{(n-1)(1-\beta_1^{1/(n-1)})(1-\beta_1)}{2(1+\beta_1)} < \frac{(-\ln\beta_1)(1-\beta_1)}{2(1+\beta_1)} < \frac{\ln 2}{6} < 1. \tag{39}$$

To establish stability, write $y_t = y_* + u_t$. Expansion around either fixed point yields

$$u_{t+1} = \big[(n-1)(1+b) - (n-2)a\big]u_t - bu_{t-1} + O(u_t^2), \tag{40}$$

with characteristic equation

$$\lambda^2 - \big[(n-1)(1+b) - (n-2)a\big]\lambda + b = 0. \tag{41}$$

The roots satisfy $|\lambda_\pm| < 1$ if and only if

$$b < 1, \qquad 0 < (n-2)(a-1-b) < 2(1+b). \tag{42}$$

Within the stated parameter range, the second condition follows automatically from $b < 1$. Indeed,

$$q > \frac{1}{\sqrt{2}}, \qquad 1-q < \frac{\ln 2}{n}, \qquad 0 < 1-bq < \frac{1}{2}, \tag{43}$$

so

$$0 < (n-2)(a-1-b) = (n-2)\frac{(1-q)(1-bq)}{q} < \frac{n-2}{n}\frac{\ln 2}{\sqrt{2}} < 2(1+b). \tag{44}$$

Conversely, $|\lambda_\pm| < 1$ requires $b = \lambda_+\lambda_- < 1$. At $b = 1$, the fixed points still exist and the same bound gives $0 < (n-2)(a-1-b) < 4$, placing the conjugate roots on the unit circle. Immediately beyond this boundary, at least one root has modulus greater than one.

It remains to transfer this stability from the limiting recurrence to the exact time-dependent one. Let $z_t = (y_t, y_{t-1})$, let $z_* = (y_*, y_*)$, and let A be the Jacobian of the limiting two-dimensional map at z_*. When $b < 1$, its spectral radius is less than one, so there is an equivalent norm and a number $\rho < 1$ for which the limiting nonlinear map is locally Lipschitz with constant at most ρ around z_*. Because $a_t - a$, $b_t - b$, and $c_t - c$ are all $O(q^{2t})$, the exact recurrence satisfies, after shrinking the neighborhood if necessary,

$$\|z_{t+1} - z_*\| \le \rho\|z_t - z_*\| + Cq^{2t}. \tag{45}$$

Iteration gives

$$\|z_t - z_*\| \le \rho^{t-T}\|z_T - z_*\| + C\sum_{s=T}^{t-1}\rho^{t-1-s}q^{2s} \longrightarrow 0 \tag{46}$$

for every trajectory entering this local basin at a sufficiently large time T. Thus the summable perturbation from gradient replenishment does not change the local stability boundary.

Thus, for $b < 1$, trajectories attracted to either fixed point satisfy $S_t \to S_* < 1$ and eventually cease crossing the EoS threshold. The loss of strict linear stability occurs at $b = 1$, giving $\beta_2 = \beta_1^{2(n-2)/n}$. □

A.3 Proof of Lemma 5

Proof. The generalized Gauss–Newton decomposition Schraudolph (2002) gives

$$\nabla_\theta^2 \ell_i = \underbrace{J_i^\top C_i J_i}_{G_i} + \underbrace{\sum_k (p_{ik} - y_{ik})\nabla_\theta^2 z_{ik}}_{R_i}, \qquad C_i = \operatorname{diag}(p_i) - p_i p_i^\top. \tag{47}$$

Let c_i be the correct class and $\varepsilon_i = 1 - p_{i,c_i}$. Since $C_i \succeq 0$,

$$\|C_i\|_2 \le \operatorname{tr} C_i = 1 - \|p_i\|_2^2 \le 1 - (1-\varepsilon_i)^2 \le 2\varepsilon_i, \qquad \|p_i - y_i\|_1 = 2\varepsilon_i. \tag{48}$$

Writing $B_{1,i} = \|J_i\|_2$ and $B_{2,i} = \max_k \|\nabla^2_\theta z_{ik}\|_2$, submultiplicativity and the triangle inequality yield

$$\|G_i\|_2 \le 2\varepsilon_i B_{1,i}^2, \qquad \|R_i\|_2 \le 2\varepsilon_i B_{2,i}, \qquad \|\nabla^2_\theta L\|_2 \le \frac{2}{N}\sum_i \varepsilon_i (B_{1,i}^2 + B_{2,i}) \longrightarrow 0. \quad (49)$$

The same argument applies to coordinate-wise binary cross-entropy. For a binary target $y_{ik} \in \{0, 1\}$, define the signed margin $m_{ik}(s) = (2y_{ik} - 1)z_{ik}(s)$. Then

$$L_{\mathrm{BCE}}(s) = \frac{1}{ND}\sum_{i,k} \log\left(1 + e^{-m_{ik}(s)}\right), \quad (50)$$

which is a two-class cross-entropy for each sample–coordinate pair. Hence, if every sigmoid prediction approaches its binary target and the corresponding logit derivatives remain bounded, the preceding Hessian-collapse bound holds after replacing i by the compound index (i, k). Moreover, in the confident regime, locally linearizing $m_{ik}(s) \simeq m_{ik} - a_{ik}s$ gives $L_{\mathrm{BCE}}(s) \simeq (ND)^{-1}\sum_{i,k} e^{-m_{ik}+a_{ik}s}$, exactly the exponential-sum form in Definition 6. If reconstruction remains imperfect so that some pixel probabilities do not approach their targets, however, the vanishing-Hessian conclusion need not hold and a nonzero-curvature core may remain. □

A.4 Proof of Theorem 7

Proof. The ratio $L_{\mathrm{conf}}(s)/L_{\mathrm{conf}}(0)$ is the moment-generating function of the CE-weighted slope distribution. Replacing this distribution by the Gaussian model gives

$$\frac{L_{\mathrm{conf}}(s)}{L_{\mathrm{conf}}(0)} = \mathbb{E}_w[e^{as}] = \exp\left(\mu_a s + \frac{\sigma_a^2 s^2}{2}\right). \quad (51)$$

Because $s = 0$ is the local minimum of the slice,

$$0 = \frac{L'_{\mathrm{conf}}(0)}{L_{\mathrm{conf}}(0)} = \mathbb{E}_w[a] = \mu_a. \quad (52)$$

Therefore, with $u = \sigma_a s$,

$$\frac{\Delta L_{\mathrm{conf}}(s)}{L_{\mathrm{conf}}(0)} = e^{u^2/2} - 1. \quad (53)$$

Define the effective log–log exponent by

$$n_{\mathrm{eff}}(s) = \frac{\mathrm{d}\log \Delta L_{\mathrm{conf}}(s)}{\mathrm{d}\log |s|}. \quad (54)$$

The Gaussian model gives

$$n_{\mathrm{eff}}(u) = \frac{u^2 e^{u^2/2}}{e^{u^2/2} - 1}. \quad (55)$$

For $|u| \ll 1$,

$$n_{\mathrm{eff}}(u) = 2 + \frac{u^2}{2} + O(u^4). \quad (56)$$

Hence the loss is quadratic for $|u| \ll 1$ and departs from the quadratic form when $|u| = O(1)$. Since $u = \sigma_a s$, the quadratic-core scale is $s_{\mathrm{core}} = O(1/\sigma_a)$. Defining the core boundary by $n_{\mathrm{eff}} = 2.5$ gives $|u_{\mathrm{core}}| \simeq 0.964$, or

$$|s_{\mathrm{core}}| \simeq \frac{0.964}{\sigma_a}. \quad (57)$$

The same derivation applies to confident coordinate-wise BCE because its signed-margin approximation is the same exponential sum over sample–coordinate pairs. □

B FULL-MODEL EXPERIMENTS

B.1 EXPERIMENTAL SETUP FOR THE SIX MODELS

All six beta-plane scans use a fixed random seed of 42 and AdamW. The beta values are varied across the scan and are therefore omitted from the setup below. The training horizon is adapted for each task: each run is inspected at successive stages, and unresolved points are resumed at a longer horizon when necessary. The maximum horizon reported here is the largest extension used or permitted for the corresponding final panel, rather than a fixed number of steps applied to every beta setting. Parameter counts refer to trainable model parameters and exclude optimizer state.

B.1.1 MODULAR DIVISION

Model. The modular-division model is a one-layer causal Transformer with vocabulary size 54, model dimension $d_{\text{model}} = 128$, four attention heads, head dimension $d_{\text{head}} = 32$, and an MLP dimension of 512. The sequence length is three, corresponding to the prompt $[x, y, =]$. The model uses ReLU activations and no LayerNorm. The input embedding, positional embedding, attention, MLP, and output unembedding together contain 211,456 trainable parameters.
Dataset. The task is $xy^{-1} \bmod 53$, with $x \in \{0, \ldots, 52\}$ and $y \in \{1, \ldots, 52\}$. The complete dataset contains 2,756 examples, which are randomly split into 1,378 training and 1,378 test examples using a 50% training fraction.
Optimizer. Training uses cross-entropy loss, AdamW with learning rate 10^{-3}, weight decay 0.5, and Adam epsilon 10^{-8}. The training loader uses batch size 512, and the learning rate is linearly warmed up during the first 100 optimizer steps.
Training steps. The final scan starts from 8,000 optimizer steps and adaptively extends unresolved boundary candidates, with a maximum horizon of 30,000 steps.

B.1.2 MODULAR ADDITION

Model. The modular-addition model uses the same one-layer Transformer architecture as modular division: vocabulary size 54, $d_{\text{model}} = 128$, four heads with $d_{\text{head}} = 32$, MLP dimension 512, sequence length three, ReLU activations, no LayerNorm, and 211,456 trainable parameters.
Dataset. The task is $(x + y) \bmod 53$, with $x, y \in \{0, \ldots, 52\}$. The complete dataset contains 2,809 examples, randomly split into 1,404 training and 1,405 test examples using a 50% training fraction.
Optimizer. Training uses cross-entropy loss, AdamW with learning rate 10^{-3}, weight decay 0.5, and Adam epsilon 10^{-8}. The batch size is 512, with the same linear 100-step learning-rate warmup as modular division.
Training steps. The final scan starts from 8,000 optimizer steps and adaptively extends unresolved boundary candidates up to a maximum of 30,000 steps.

B.1.3 TINY SHAKESPEARE

Model. The Tiny Shakespeare model is a compact causal character Transformer. It has a 65-character vocabulary, model dimension 48, four attention heads, two Transformer blocks, and an MLP dimension of 192 in each block. The character and positional embeddings are 48-dimensional, the positional context length is 64, and the output classifier has no bias. The model uses GELU activations, zero attention dropout, and no LayerNorm. It has 65,472 trainable parameters.
Dataset. The data are drawn from the first 90% of `data/tinyshakespeare/input.txt`, giving a training corpus of 1,003,854 characters. Each run uses the same fixed random set of 448 character windows, each with context length 64. Only the final eight target positions in each window contribute to the loss, while the preceding 56 target positions are masked. The loss is cross-entropy evaluated in float64.
Optimizer. Training uses full-batch AdamW with learning rate 10^{-3}, weight decay 0.1, and Adam epsilon 10^{-8}, with no learning-rate scheduler.
Training steps. The base trajectory is 12,000 optimizer steps. Unresolved points are adaptively extended through 30,000 and 50,000 steps, with selected points extended to a maximum of 100,000 steps.

B.1.4 CIFAR-10 MLP

Model. The CIFAR-10 MLP receives a $3 \times 32 \times 32$ image, flattened to 3,072 input features. It contains five hidden linear layers of width 512, with a ReLU after each hidden layer, followed by a

linear 512-to-10 classifier. The model has 2,629,130 trainable parameters and does not use dropout or normalization layers.
Dataset. For each run, 200 images are randomly selected from the CIFAR-10 training set and 200 images from the CIFAR-10 test set, using seed 42. The images are converted with `ToTensor()` only, with no data augmentation or additional normalization.
Optimizer. Optimization uses the 200 training images in full batch with cross-entropy loss and AdamW, using learning rate 10^{-3}, weight decay 0.5, and Adam epsilon 10^{-8}.
Training steps. The final scan starts from 4,000 optimizer updates per run and adaptively extends unresolved points to a maximum of 16,000 updates.

B.1.5 CIFAR-10 CNN

Model. The CIFAR-10 CNN is a VGG11-style convolutional network adapted to 32×32 inputs. Its eight convolutional layers have channel widths $3 \to 64 \to 128 \to 256 \to 256 \to 512 \to 512 \to 512 \to 512$, with ReLU activations and five 2×2 max-pooling operations. The classifier maps the final 512 features through two 512-unit linear layers to 10 classes. Each of the two intermediate classifier layers is followed by ReLU and dropout with probability 0.5. The model has 9,750,922 trainable parameters.
Dataset. The data protocol is the same as for the CIFAR-10 MLP: 200 randomly selected CIFAR-10 training images and 200 randomly selected test images, `ToTensor()` only, and no augmentation or additional normalization.
Optimizer. Optimization is full-batch cross-entropy training with AdamW, learning rate 10^{-3}, weight decay 0.5, and Adam epsilon 10^{-8}.
Training steps. The base scan uses 8,000 optimizer updates per run. Unresolved points are adaptively extended to 16,000 or 30,000 updates, with a maximum horizon of 30,000 updates.

B.1.6 MNIST AUTOENCODER

Model. The MNIST model is a compact binary convolutional autoencoder. The encoder consists of a $1 \to 8$ convolution and an $8 \to 16$ convolution, both with kernel size 3, stride 2, and ReLU activations, reducing a 28×28 image to a $16 \times 7 \times 7$ representation. A linear layer maps this representation to a 32-dimensional latent vector, and a second linear layer maps it back to $16 \times 7 \times 7$. The decoder uses a $16 \to 8$ transposed convolution followed by ReLU and an $8 \to 1$ transposed convolution that produces the output logits. Both transposed convolutions use kernel size 4 and stride 2. The autoencoder has 54,425 trainable parameters.
Dataset. Each run uses 64 randomly selected images from the MNIST training split, with seed 42. The grayscale inputs are binarized at threshold 0.5, and the reconstruction target is the binarized input itself.
Optimizer. Training uses binary cross-entropy with logits and full-batch AdamW with learning rate 2×10^{-3}, weight decay 0.2, and Adam epsilon 10^{-12}. No learning-rate scheduler is used.
Training steps. The initial phase scan checks successive horizons, and unresolved points are adaptively resumed through a maximum of 200,000 optimizer steps.

B.2 PERIOD-DETECTION ALGORITHM

We estimate the macroscopic period of loss oscillations from loss traces sampled once per optimizer step. We use the training loss for modular arithmetic, Tiny Shakespeare, and MNIST reconstruction, and the test loss for the two CIFAR-10 models. For a trajectory of S steps, we discard the first 20% to reduce the influence of the initial transient and analyze the remaining segment. Segments containing fewer than 100 samples are not assigned a period.

To suppress rapid fluctuations while accommodating the large dynamic range of the loss, we first transform the signal as $u_t = \log_{10}(L_t + 10^{-12})$ and smooth it with a third-order Butterworth low-pass filter applied forward and backward. The cutoff frequency is 0.05 cycles per step for all models. We then restore the linear loss scale, $\widetilde{L}_t = 10^{\widetilde{u}_t}$, and subtract its mean. We estimate the power spectrum $P(f)$ of this signal using Welch's method.

We identify local spectral peaks and retain only those satisfying both

$$\text{prominence}(f_k) \geq 0.02 \max_f P(f), \qquad f_k > \frac{1}{T_{\max}}, \qquad T_{\max} = S. \tag{58}$$

Here, prominence measures how strongly a spectral peak stands out from its surrounding spectral background, and $T_{\max}$ is the length of the complete trajectory before transient removal. Among the accepted peaks, we select the one with the largest spectral power and define the detected period as

$$f_* = \arg\max_{f_k \in \mathcal{C}} P(f_k), \qquad T = \frac{1}{f_*}, \tag{59}$$

where $\mathcal{C}$ denotes the set of accepted peaks. The resulting T is measured in optimizer steps. If no peak satisfies both criteria, the trajectory is marked as having no detected period within the observation window. Unresolved scan points are adaptively extended as described in Appendix B.1, and the period is re-estimated from the longer trace. For all models except the CIFAR-10 MLP, we additionally require the minimum training loss in the final 10% of the complete trajectory to be at most 0.05. We exempt the CIFAR-10 MLP because dead ReLU units can prevent its training loss from reaching this threshold even when the trajectory remains relevant to the phase-boundary analysis.

C One-dimensional toy-model controls

C.1 Sensitivity of the Adam boundary for $L(x) = kx^2/2$

To test the sensitivity of the quadratic toy-model boundary to non-beta Adam parameters, we use a common grid spanning $\delta = 1 - \beta_1 \in [10^{-2}, 10^{-1}]$ and $\gamma = 1 - \beta_2 \in [5 \times 10^{-4}, 10^{-1}]$. Each control changes one of k, η, or ϵ relative to the baseline in Figure 6(a), while holding the other two fixed. As in the main quadratic experiment, each beta pair is trained for 250,000 updates, the first 125,000 updates are discarded, and a period is assigned when at least three upward crossings of the local stability threshold are detected. We fit the resulting boundary to $\gamma = C\delta^p$. Figure 6 shows the phase diagrams, and Table 2 reports the fits.

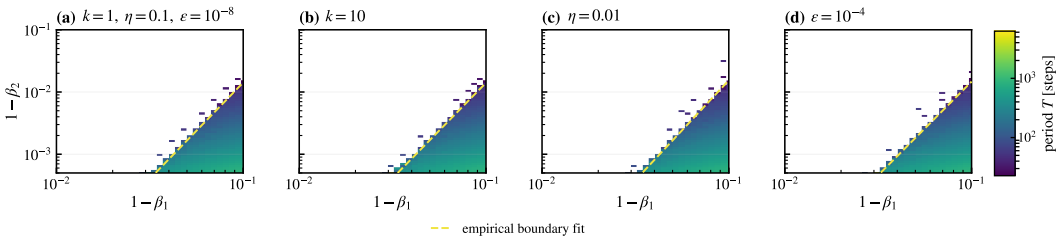

Figure 6: Sensitivity of the quadratic-loss boundary to k, η, and ϵ. The phase diagrams show $L(x) = kx^2/2$ in the $(1-\beta_1, 1-\beta_2)$ plane, with dashed fits $\gamma = C\delta^p$.

Table 2: Power-law fits for the quadratic-loss parameter controls in Figure 6.

Panel	Parameters	Valid points	p	C
(a)	$k = 1, \eta = 0.1, \epsilon = 10^{-8}$	351	3.095	17.70
(b)	$k = 10, \eta = 0.1, \epsilon = 10^{-8}$	353	3.064	16.30
(c)	$k = 1, \eta = 0.01, \epsilon = 10^{-8}$	351	3.210	25.00
(d)	$k = 1, \eta = 0.1, \epsilon = 10^{-4}$	365	3.009	15.02

Changing k by one order of magnitude leaves the fitted boundary nearly unchanged, and changing η preserves the approximately cubic scaling while producing only a moderate shift in its prefactor. The robustness to ϵ is conditional: ϵ must remain small compared with $\sqrt{\hat{v}_t}$ over the part of the trajectory that determines the boundary. If ϵ is too large, it dominates the denominator $\sqrt{\hat{v}_t} + \epsilon$ and suppresses the late-time decay of the effective second-moment scale, so the expected boundary scaling is no longer obtained.

C.2 Effect of weight decay in the one-dimensional toy model

Weight decay contributes the parameter update $\Delta\theta_{\text{wd}} = -\eta\lambda_{\text{wd}}\theta$. In a neural network, the parameter origin $\theta = 0$ and a local minimum of the task loss are generally distinct. Moreover, the weight-decay update and the Adam-preconditioned gradient direction tend to have small overlaps in high dimensions Cai et al. (2013). At the modular-division pre-spike point used in Figures 1(c) and 5(a), their signed Euclidean projection is

$$\left\langle \Delta\theta_{\text{wd}}, \frac{\hat{d}_t}{\|\hat{d}_t\|_2} \right\rangle = +9.450 \times 10^{-3}. \tag{60}$$

This geometry cannot be represented by the centered one-dimensional loss $L(x) = k|x|^n/n$: in one dimension, weight decay and the loss gradient can only be parallel or antiparallel, and decay toward the origin also points toward the loss minimum at $x = 0$. We therefore shift the minimum and use

$$L(x) = \frac{k}{n}|x - x_*|^n, \qquad x_* = 1. \tag{61}$$

We choose the small value $\lambda_{\text{wd}} = 10^{-3}$ so that the projected decay term perturbs rather than dominates the Adam dynamics. We set $x_0 = 2$, preserving the unit initial displacement $x_0 - x_* = 1$, and otherwise retain the main toy-model settings: $k = 1$, $\eta = 0.1$, $\epsilon = 10^{-30}$, 250,000 updates, and a 125,000-update burn-in.

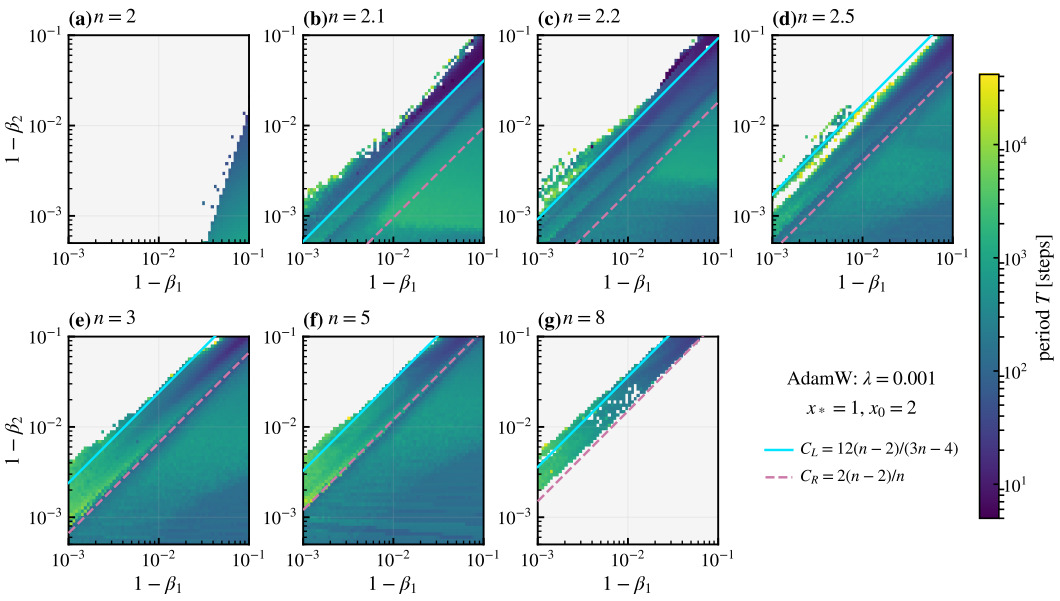

Figure 7: Effect of weak decoupled weight decay on the shifted one-dimensional losses $L(x) = |x - 1|^n/n$. AdamW uses $\eta = 0.1$, $\lambda_{\text{wd}} = 10^{-3}$, $\epsilon = 10^{-30}$, and $x_0 = 2$. Each beta pair is run for 250,000 updates, with the first 125,000 discarded. Color indicates the period estimated from repeated upward crossings of the local stability threshold. The solid cyan and dashed pink lines show $\gamma = 12(n-2)\delta/(3n-4)$ and $\gamma = 2(n-2)\delta/n$, respectively. Relative to Figure 3, the left boundary shifts slightly upward, periodic trajectories extend below the reference right boundary, and the right edge of the wedge is unresolved.

The weak shifted decay therefore preserves the approximately unit-slope left boundary while modestly enlarging the periodic region on that side. However, the lower/right boundary that closes the no-decay wedge disappears over the scanned range. Thus, the near-linear left-boundary scaling is robust to this weak projected weight-decay force, whereas the two-sided wedge is not.

D FINITE-SCALE LANDSCAPE ANALYSIS

D.1 LOCATING THE PRE-SPIKE POINTS AND PROGRESSIVE SHARPENING AT FINITE SCALES

We define a temporal valley as a low-loss segment between macroscopic high-loss excursions. After discarding the initial training transient, the loss trace is partitioned into successive valleys using the task's loss scale: a valley must enter the low-loss regime and must subsequently be closed by a new high-loss excursion. The last valley in a finite trace is marked as right-censored if no closing excursion has yet been observed and is not used in the exponent fit.

Within each completed valley V_j, the pre-spike point is the single checkpoint

$$t_{\mathrm{pre},j} = \arg\min_{t \in V_j} L_t. \tag{62}$$

Thus each completed valley contributes at most one event, preventing long valleys or densely saved portions of a trace from receiving disproportionate weight. The pre-spike point is selected solely from the temporal loss trace. No directional-slope or landscape-shape criterion is used to move the anchor to a nearby checkpoint. When the exact pre-spike point was not retained as a checkpoint during the original run, deterministic replay is used to reconstruct the model and optimizer state at $t_{\mathrm{pre},j}$.

Across the examined models and β settings, the selected points occur immediately before a spike. Figure 8 illustrates the selection for three modular-division trajectories.

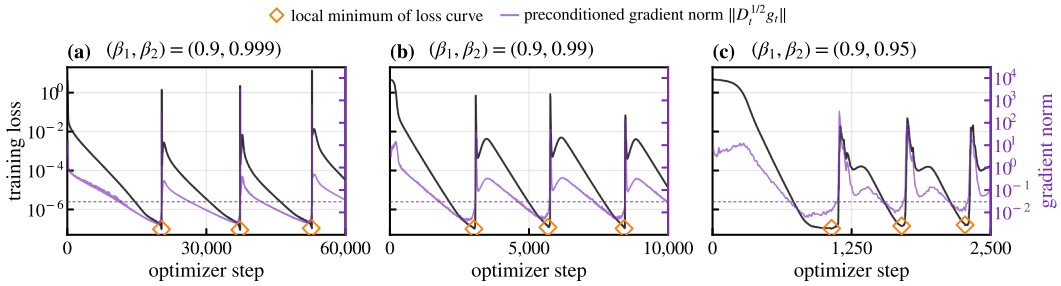

Figure 8: Training-loss traces for the modular-division Transformer at $(\beta_1, \beta_2) = (0.9, 0.999)$, $(0.9, 0.99)$, and $(0.9, 0.95)$, from left to right. Diamonds mark the local minima of the loss curve, which are used directly as the pre-spike points.

The pre-spike point t_{pre} and the spatial minimum s_* play different roles. The former selects the pre-spike model state. After freezing that state and the Adam preconditioner, we define the full-model directional slice

$$\phi_t(s) = L(\theta_t + s\hat{d}_t), \qquad \hat{d}_t = \frac{D_t g_t}{\|D_t^{1/2} g_t\|_2}, \tag{63}$$

and refine its one-dimensional minimum s_*. The direction combines the current gradient with the second-moment preconditioner and captures the geometry relevant to the Adam update. The exponent fit is centered at s_*, so the procedure does not assume that the pre-spike point itself lies exactly at a stationary point of the frozen directional slice. To compare the slice with the dynamics, we freeze $\hat{d}_t$ at step t. For the ten updates ending at t, let $\Delta\theta_j = \theta_j - \theta_{j-1}$. We calculate their projected slice coordinates and the typical update scale as

$$s_j^{\mathrm{proj}} = \frac{\Delta\theta_j^\top \hat{d}_t}{\hat{d}_t^\top \hat{d}_t}, \qquad \overline{\Delta s} = \operatorname*{median}_{j=t-9,\ldots,t} \left| s_j^{\mathrm{proj}} \right|. \tag{64}$$

The denominator expresses each projected update in the same s coordinate used by $\phi_t(s)$, even though $\hat{d}_t$ is not Euclidean-normalized.

We classify each recentered slice according to Definition 4. We use the modular-division Transformer at $(\beta_1, \beta_2) = (0.9, 0.999)$ as a representative example.

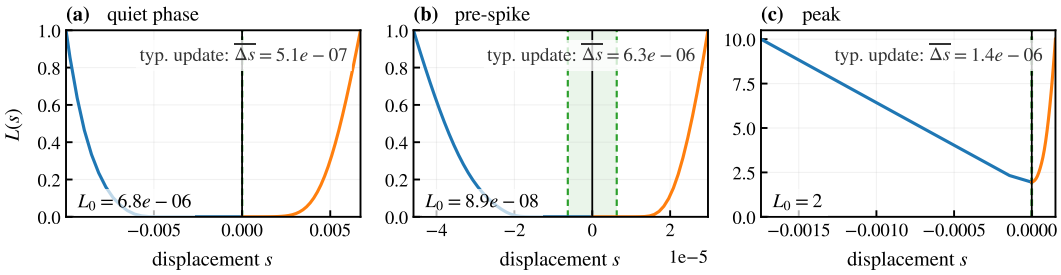

Figure 9: Evolution of the directional loss slice through a modular-division spike at $(\beta_1, \beta_2) = (0.9, 0.999)$. The three panels show a quiet point (step 28,600, $L_0 \simeq 6.8 \times 10^{-6}$), the pre-spike point (step 37,210, $L_0 \simeq 8.9 \times 10^{-8}$), and the spike peak (step 37,220, $L_0 \simeq 2.0$), respectively. The green band and dashed edges indicate the magnitude of a typical Adam-preconditioned update: $\overline{\Delta s} \simeq 5.1 \times 10^{-7}$, 6.3×10^{-6}, and 1.4×10^{-6} in the three panels. As the system approaches the pre-spike state, the flat core contracts to the update scale. **This phenomenon can be viewed as progressive sharpening at finite scales.** At the peak, the directional wall is encountered and the slice becomes strongly nonquadratic.

We define the core–wall transition as the first displacement at which

$$n_{\text{eff}}(s) = \frac{\mathrm{d} \log \Delta L}{\mathrm{d} \log |s - s_*|} \tag{65}$$

reaches 2.5. During the quiet phase in Figure 9, the left and right transition distances are approximately 7.0×10^{-4} and 8.0×10^{-4}, more than three orders of magnitude larger than $\overline{\Delta s} \simeq 5.1 \times 10^{-7}$. At the pre-spike point, these distances contract to 2.16×10^{-6} and 3.55×10^{-6}, while $\overline{\Delta s}$ grows to 6.3×10^{-6}. **A typical update therefore crosses the quadratic core and reaches the superquadratic wall.** At the spike peak, the reference loss is approximately 2.0, and the strongly asymmetric slice no longer exhibits a core–wall landscape.

D.2 Fitting and aggregating the wall exponent

Every model–task setting is evaluated at the same six reference beta pairs,

$$(\beta_1, \beta_2) \in \{(0.9, 0.9), (0.9, 0.95), (0.9, 0.99), (0.9, 0.999), (0.99, 0.99), (0.99, 0.999)\}. \tag{66}$$

For each usable pre-spike slice, define the displacement and excess loss on its left and right branches by

$$r = |s - s_*|, \qquad \Delta L = \phi_t(s) - \phi_t(s_*), \tag{67}$$

and transform the positive, resolved samples to $u = \log r$ and $y = \log \Delta L$. The two branches are retained separately so that basin asymmetry is absorbed by branch-specific parameters rather than by averaging the profiles before fitting.

Each event-side branch is first fit with a continuous three-segment line,

$$y(u) = c + n_{\text{core}}(u - b_1) + (n_{\text{wall}} - n_{\text{core}})[u - b_1]_+ + (n_{\text{roll}} - n_{\text{wall}})[u - b_2]_+, \qquad [z]_+ = \max(z, 0). \tag{68}$$

The breakpoints $b_1 < b_2$ separate the quadratic core, steep wall, and outer rollover. Candidate breakpoints are enumerated, with at least five samples required in each regime, and the pair with the smallest total squared residual in log space is selected subject to

$$n_{\text{wall}} > n_{\text{core}}, \qquad n_{\text{wall}} > n_{\text{roll}}. \tag{69}$$

These inequalities operationally identify the wall as the middle steepening region. They also prevent the flatter outer rollover from being folded into the phase-relevant wall exponent.

After determining b_1 and b_2 separately for every event and side, all branches within one beta setting are refit jointly using only samples through b_2:

$$y_{e,\pm}(u) = c_{e,\pm} + n_{\text{core}} \min(u - b_{1,e,\pm}, 0) + n_{\text{wall}} \max(u - b_{1,e,\pm}, 0), \qquad u \le b_{2,e,\pm}. \tag{70}$$

The intercept and breakpoints remain specific to each event-side branch, while n_{core} and n_{wall} are shared. This produces one beta-level exponent $n_{\mathrm{wall},k}$ from all usable pre-spike events at the k-th beta pair. Uncertainty is estimated with 2,000 event-level bootstrap resamples, with the left and right branches of each event always resampled together.

Finally, the exponent shown in the main-text table is the unweighted arithmetic mean of the six beta-level estimates,

$$n_{\mathrm{wall}}^{(\mathrm{task})} = \frac{1}{6}\sum_{k=1}^{6} n_{\mathrm{wall},k}. \tag{71}$$

Each beta pair therefore receives the same weight even when the number of usable completed valleys differs. The boundary coefficient is computed only after this averaging step,

$$C_L = \frac{12\big(n_{\mathrm{wall}}^{(\mathrm{task})} - 2\big)}{3n_{\mathrm{wall}}^{(\mathrm{task})} - 4}, \tag{72}$$

rather than by averaging six separately transformed coefficients.