STABILITY OF DEEP TEST-TIME-TRAINING MEMORIES: DIVERGENCE AND SILENT COLLAPSE

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time-training (TTT) memories, introduced by Titans and ATLAS, store the past in the weights of a small neural network that keeps learning as the model runs. Each input triggers a gradient step on the memory's own prediction error. The memory updates itself on the fly at a fixed cost. This made long-term memory practical at scale and started a fast-growing line of work. The mechanism is delicate. The memory is a network continuously rewritten by the data it must retain, and its update is repeatedly reported to be unstable. No condition is known for when the update stays bounded. None is known for whether the trained memory stays healthy. For a memory that is itself a deep nonlinear network, neither has been established. We study both questions on two independent platforms and find two mechanisms. *First, divergence.* An update changes the network so that the next update is larger, and the state blows up. We give a condition that a practitioner can check and that prevents it: bound the size of each write, and keep a positive forgetting rate. Orthogonalising the write direction without bounding its size does not prevent it. *Second, collapse.* The memory stays numerically stable but loses its diversity. Nearly every write ends up pointing the same way. Collapse is learned during training, and the prediction loss barely moves while it happens. We give the tools to catch both failures: the stability condition above, and a paired retention probe with a rank diagnostic that read memory diversity directly. Whether collapse costs accuracy at scale is open. What we establish is that the prediction loss will not tell you it happened.

1 INTRODUCTION

Sequence models are limited by how much of the past they can use. Transformers keep every token in an attention cache that grows with the context, so the total cost of reading it is quadratic. This is prohibitive for long documents, hours of video, or an agent that must remember across a long interaction.

Test-time-training (TTT) memories take a different route. Instead of storing the past explicitly, they compress it into the weights of a small network that keeps learning while the model runs. Each new input triggers a gradient step on the memory's own prediction error, so the memory updates itself on the fly and its size never grows. This is the idea behind Titans (Behrouz et al., 2025b) and ATLAS (Behrouz et al., 2025a), which generalise the fast-weight view of linear attention (weights the forward pass rewrites, unlike the slow weights fixed in pre-training) (Katharopoulos et al., 2020; Schlag et al., 2021) and can be interpreted as an in-context gradient descent (von Oswald et al., 2023). This has made fixed-cost long-term memory practical at scale, and inspired a fast-growing line of work on test-time learning and memory (Behrouz et al., 2026; von Oswald et al., 2026; Li et al., 2026; Zhang et al., 2026).

The idea is attractive, but the mechanism is delicate. The memory is a network continuously rewritten by the data it must retain, and the online update can go wrong. The literature says so repeatedly. Deep memories fail to converge (Park et al., 2025), training is reformulated to stay numerically stable (von Oswald et al., 2026), and long-context runs are prone to gradient explosion (Tandon et al., 2025). The most-used open-source implementation (Wang, 2025) adds three stabilising patches beyond the published rule and still reaches non-finite states in our experiments. None of the original

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.

papers gives a condition under which this online update stays stable, and for a memory that is itself a deep, multi-layer nonlinear network none has been established (Section 2).

This paper addresses that gap. We ask two questions. When does the online update stay bounded? And once it is bounded, does the memory stay healthy, or can it degrade unseen? We answer the first with a stability condition. We answer the second by finding the cause of collapse and by giving a way to measure it. Both answers are practical. A practitioner can check the stability condition. A practitioner can watch memory diversity during training instead of trusting the loss. Whether collapse also costs accuracy at scale is open. We do not settle whether it is a pathology or a rational bypass of a module the task does not yet need.

We find two failure modes. **First, divergence:** the update feeds on itself and the state blows up inside a single pass (§4). **Second, collapse:** the memory stays numerically fine, but slowly loses its write diversity, and the outer training loop learns to make this happen (§5), a learned optimiser reaching a degenerate regime (Metz et al., 2019). Collapse is silent. Prediction loss barely moves while it happens, so a memory can look healthy by the metric that trains it while its writes have concentrated in a single direction. Both failures act through the writes, divergence through their magnitude, collapse through their diversity.

The prediction loss does not separate a healthy memory from a collapsed one. A memory that stays active while it collapses reads the same to the loss that trains it. If a diverse memory matters for the task, the loss will not report when that diversity is lost. Memory diversity therefore has to be measured directly.

This work prioritises mechanism and causal analysis over benchmark performance. We study a family of update rules, reproduce each effect on two independent platforms, use interventions to separate mechanism from correlation, and judge the memory by what it can recall, not by its prediction loss. The aim is a practical, mechanistic account of how these memories fail and how to keep them healthy.

Contributions.

- We show that divergence is a feedback loop between the size of each write and the curvature of the memory's inner loss, and we give its exact stability region in a simplified setting. This gives a design rule you can apply. Bound the size of each write. The memory then stays inside a region set by the write size and the forgetting rate, the two knobs that also set its memory span. In the rules we study, stability comes from that bound, not from the orthogonalisation. ATLAS's write is also magnitude-bounded, so we read its stability the same way.
- We find the cause of silent collapse on our testbed. It is the momentum gate, which suppresses the effective writes. Moving that one learned component between a healthy and a collapsed memory switches the collapse off or on. Repairing the geometry after training does not repair the function, because the rest of the network has co-adapted to the collapsed writes. The same collapse appears on the most-used open-source implementation, where it comes instead from near-amnesic forgetting.
- Because the prediction loss reports neither divergence nor collapse, we give a memory-specific check: a paired retention probe and a rank diagnostic that read memory diversity and retention directly. They are cheap, and they drop into a training run as a health signal the loss does not provide.

2 Related work

Writing into fast weights by gradient descent is an old idea (Schmidhuber, 1992). It came back through the reading of linear attention as a fast-weight memory (Schlag et al., 2021), and then as test-time training (Sun et al., 2020; 2024). Titans (Behrouz et al., 2025b) added the gated update we study; ATLAS (Behrouz et al., 2025a) added an orthogonalised write taken from the Muon optimiser (Jordan et al., 2024); MIRAS (Behrouz et al., 2026) recasts the family as online optimisation with a retention term. Work on the systems side is active (von Oswald et al., 2026; Zhang et al., 2026; Li et al., 2026; Tandon et al., 2025), and the instability we study runs through it.

Stability is understood for linear and structured memories, where the update is an explicit matrix and its spectrum can be controlled (Orvieto et al., 2023; Yang et al., 2024; 2025b; Siems et al., 2025; Liu et al., 2024). Preconditioned DeltaNet (Tumma et al., 2026) gives the sharpest such treatment, but sets the Titans and ATLAS family aside because its update is nonlinear in the state. For that deep, nonlinear case we found no published stability condition (Appendix H). A recent reimplementation re-examines Titans empirically (Di Nepi et al., 2025) without deriving one. The latest works do not close the gap. MIRAS (Behrouz et al., 2026) uses the word stability for a property of its objective, the retention term, not for whether the update diverges. Large-chunk TTT (Zhang et al., 2026) keeps its updates stable in practice through normalisation, without asking when they diverge. Modular TTT (Tang et al., 2026) comes closest. For a shallow, momentum-free fast weight it derives a spectral condition ($\lambda_{\max}(H) < 2$ on the update map $I - H$, held there by a small learning rate), the momentum- and forgetting-free corner of our region, since Proposition 1 at $\eta = \alpha = 0$ reads $\theta\lambda < 2$. It also treats the deep case, but as first-order update *geometry* (each factor coupled through the others on its forward–backward path), and reports empirically that deeper fast weights hurt by inducing large activations. Two things it leaves open are what we add. First, it analyses no curvature, so it misses that the deep memory rewrites its own curvature as it writes. That self-reinforcing loop is the one of which their large-activation finding is a symptom. Second, it treats no momentum or forgetting, the gates whose stability region Proposition 1 gives. We keep the deep, gated case and stabilise it by sizing the writes rather than by keeping the memory shallow.

Rank collapse is known in other settings, from deep attention stacks to long-context recurrent states (Chen et al., 2025) to continually trained networks; Appendix H gives the parallels and where they differ. In the linear-attention case, Nazari & Rusch (2026) link a reduced state rank to degraded long-range retrieval. Our case is new in that the collapsing object is a fast weight matrix written at test time, and the outer loop learns the gates that keep it degenerate while the loss stays flat. For evaluation, recall probes (Arora et al., 2024) share our view that prediction loss under-reports memory. Our paired protocol goes one step further and cancels the generic benefit of writing anything, so what remains is what the memory actually kept.

3 SETTING

The memory is a two-layer network $\mathcal{M}_W(k) = W_2\,\sigma(W_1 k)$ whose *fast* weights $W = (W_1, W_2)$ are rewritten online. Learned projections map each latent to a key k_t, a value v_t and a query. The memory is trained on the fly on the inner loss $\ell_t(W) = \|\mathcal{M}_W(k_t) - v_t\|^2$, through the gated accumulate-and-write recurrence of Titans:

$$S_t \;=\; \eta_t S_{t-1} \;-\; \textstyle\sum_j g_j, \qquad W_t \;=\; (1-\alpha_t)W_{t-1} \;+\; U_t. \tag{1}$$

Here t indexes tokens and the g_j are inner-loss gradients taken at the weights frozen at the current chunk's start, one per step. The buffer S_t accumulates them, and the *write map* U_t turns the buffer into an update. The momentum $\eta_t \in [0, 1)$, forgetting $\alpha_t \in [0, 1]$ and step $\theta_t \geq 0$ are small learned networks, the *gates*, that read the current input; θ_t and η_t enter through the write map U_t, defined by the rules below.

We fix the recurrence (1); our object of study is the write map $S_t \mapsto U_t$. We study the maps whose magnitude is bounded by a design constant, through the saturation profile $\psi_c(r) = \frac{2}{\pi}\arctan(\pi r/2c) \in [0, 1)$, and run two of them throughout.

The **clipped rule** $U_t = D_t\, g_c(\theta_t \|S_t\|_F)$, with $g_c(x) = \frac{2c}{\pi}\arctan(\pi x/2c) \leq c$, has $\|U_t\|_F \leq c$ exactly, whatever the gradients and gates do; the direction D_t is either raw ($S_t/\|S_t\|_F$) or orthogonalised ($\mathrm{NS}(S_t)$, normalised). It is the rule the state-size lemmas of Appendix C describe verbatim.

The **Newton–Schulz rule** $U_t = \theta_t\, \psi_c(\|S_t\|_F)\, \mathrm{NS}(S_t)$ takes its orthogonalisation from the Muon optimiser (Jordan et al., 2024). Its spectral norm is $\theta_t\psi_c$ times the top singular value of $\mathrm{NS}(S_t)$, which the Newton–Schulz iterate, an *approximate* polar factor, pushes toward 1 but overshoots (to ≈ 1.2 for the standard quintic); so $\|U_t\|_2 \lesssim 1.2\,\theta_t\psi_c$ and $\|U_t\|_F \lesssim 1.2\,\theta_t\psi_c\sqrt{r}$ (r the write's rank), an exact $\theta_t\psi_c$ ceiling requiring a renormalised iterate. For this rule c is not the bound but the scale coupling the write to the buffer norm $\|S_t\|$.

The unbounded end of the family is a third arm (a separately trained variant), the raw (unbounded) rule $U_t = \theta_t S_t$. The step scales the buffer after accumulation, whereas Titans scales the new

gradient inside its momentum recurrence, so for varying θ_t this arm carries an effective momentum $\eta_t\theta_t/\theta_{t-1}$ and reproduces the published rule only up to it. It is a Titans-style stand-in, not an exact reproduction. Taking $c \to 0$ recovers *unit*, a unit-norm write direction with its magnitude discarded; the ψ_c-coupled rule, the same direction with its magnitude squashed by ψ_c, is *usq*. The figures use these two names.

Evaluation on the testbed is episodic (Figure 1). We show the model one item, then a long stretch of other content, then the same item again, and measure how well it predicts the second showing. To read off what the memory kept, we compare two streams that differ only at the start: *repeat*, which opens with the item, and *cross*, which opens with something unrelated. Everything after is identical, the distractors in between and the writing throughout. The gap between the two losses, the *retention gain* Δ, is what the memory carried across; the generic benefits of writing cancel in the difference. The attention window is fixed, so only the memory can bridge the gap, and a control with the memory switched off separates its contribution from the rest of the model. We measure this in latent space rather than on a downstream task, so the number reflects the memory alone, without a decoder's reasoning or language priors.

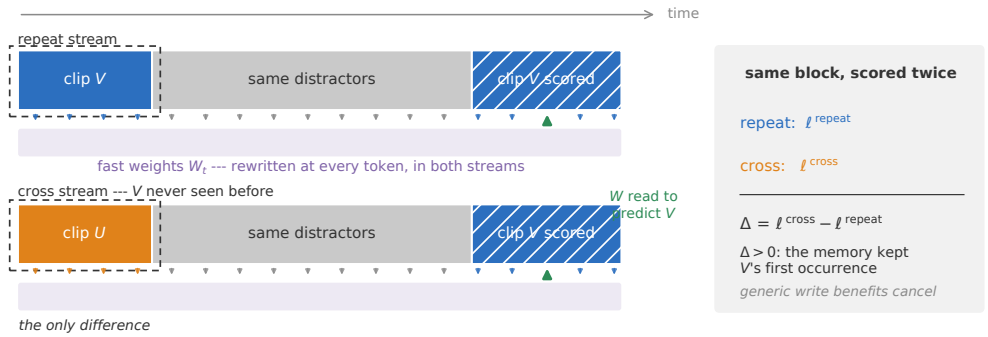

Figure 1: **The retention protocol.** Two matched streams differ only in the first block (V, or an unrelated U); the memory writes at every token of both. The gain Δ is what it kept of V.

Experiments run on two platforms. The first is a compact video world model on egocentric video, where hundreds of identically trained variants make causal interventions affordable. The second is the most-used open-source implementation of the architecture, on enwik8 language modelling, an out-of-family check. Training budgets are short and matched, $4{,}000$ steps on the reference and 10^4–3×10^4 on the testbed. Cross-arm claims compare identically trained arms; absolute losses sit far from converged values. We register predictions in writing before each campaign, run 3–5 seeds per causal claim (fewer on some divergence and frozen-momentum arms), and cost-match our comparisons, with one documented exception, the external memory-free control, which carries 12.2% fewer parameters (Appendix G).

4 First Failure Mode: Divergence

A deep test-time memory can diverge because each write's admissible step and the curvature it must stay under are the same growing quantity. It is a self-referential loop we derive, then catch in the act with the loss flat throughout.

The dynamics of (1) are driven by online gradients of the inner loss ℓ_t, modulated by the gates and the write map. For a quadratic ℓ_t a gradient step of size θ contracts iff $0 < \theta\,\lambda_{\max}(\nabla^2\ell_t) < 2$. The step-size bound ATLAS's gradient step obeys is the $\eta = \alpha = 0$ corner of this classical threshold; the gates deform the region without changing its nature. (ATLAS writes η for the step size. Here η is the momentum gate and θ the step, so their condition is our corner, not a contradiction.) The result below concerns the linear write $U_t = \theta S_t$, not the clipped or Newton–Schulz maps; chunk anchoring is treated separately in Appendix B.

Proposition 1 (quadratic stability with momentum and forgetting)**.** *Let the inner loss be quadratic with Hessian $H \succeq 0$, and use the linear write $U_t = \theta S_t$ with current-state gradients, and freeze the*

gates at constants $\theta > 0$, $\eta \in [0, 1)$, $\alpha \in [0, 1]$. The inner iteration is asymptotically stable if and only if every eigenvalue λ of H satisfies

$$-\alpha(1-\eta) \;<\; \theta\lambda \;<\; (1+\eta)(2-\alpha).$$

For $\theta > 0$ and $H \succeq 0$ the left inequality binds only at $\alpha = 0$, where a zero eigenvalue is marginally, not asymptotically, stable.

(Proof in Appendix A.) The region is wider than the classical one when momentum dominates forgetting, $\eta > \alpha/(2-\alpha)$, and narrower otherwise. The region itself is classical, that of momentum (leaky) LMS (Widrow & Hoff, 1960; Roy & Shynk, 1990) and the heavy ball (Polyak, 1964); what is specific is that the curvature (2) is written by the update. In the real system the gates and the curvature both move over time, so we do not read this region as a guarantee; we examine it as a diagnostic of divergence (Remark 2).

The proposition holds the curvature fixed, and this is where the linear and deep cases part ways. For $\mathcal{M}_W(k) = Wk$ the Hessian is $2\, k_t k_t^\top \otimes I$, with $\lambda_{\max} = 2\|k_t\|^2$, pinned by key normalisation whatever the memory writes. For the deep memory, writing $h = \sigma(W_1 k)$, the Gauss–Newton block curvatures are

$$\lambda_{\max}\big(H^{(W_2)}\big) = 2\,\|h\|^2, \qquad \lambda_{\max}\big(H^{(W_1)}\big) \;\le\; 2\,\|k\|^2\,\|W_2\,\mathrm{diag}\,\sigma'\|_2^2 \tag{2}$$

The curvature is carried by $\|h\|$ and $\|W_2\|$, quantities the memory itself rewrites at every update. (The full Hessian adds a residual $2\sum_i e_i\,\nabla_W^2 \mathcal{M}_{W,i}(k)$, $e_i = \mathcal{M}_{W,i}(k) - v_i$, with activation-second-derivative and cross-layer terms and no definite sign; we use the Gauss–Newton part as the curvature proxy throughout.) The stability condition is therefore self-referential, and predicts the failure route

$$\|W_t\| \uparrow \longrightarrow\; \lambda_{\max}\big(H_{\mathrm{GN}}(W_t)\big) \uparrow \longrightarrow\; \|\nabla \ell\| \uparrow \longrightarrow\; \|W_{t+1}\| \uparrow . \tag{3}$$

This has a falsifiable corollary. Key normalisation cannot close (3), because it bounds the wrong Gram matrix in (2). Every edge of (3) passes through the write magnitude. Bounding the writes and enforcing positive forgetting bounds the reachable state to a ball of radius $B = \max\{\|W_0\|_F,\, b/\alpha_-\}$ (Lemma 1), and that radius caps the joint Gauss–Newton curvature at $4L^2R^2B^2$; a fixed step below $(1+\eta_-)(2-\alpha_+)/(4L^2R^2B^2)$ then stays inside the region of Proposition 1 for the whole trajectory (Corollary 2, Appendix C). The loop (3) is the exploding-gradient instability of recurrent training (Pascanu et al., 2013) and an edge-of-stability regime (Cohen et al., 2021; Ahn et al., 2022; Damian et al., 2023; Arora et al., 2022) inside the memory's online update. A curvature that grows with the gradient is the non-uniform (L_0, L_1) smoothness of Zhang et al. (2020), where clipping the step succeeds and a fixed step fails, and bounding the write is that clipping applied inside the memory's own update. What is specific here is that the memory rewrites its own curvature, so crossing a local threshold is not itself a proof of nonlinear divergence, and we observe the loop directly.

The existence of a step threshold is not the finding, since every gradient method has one, and the region is nonempty (Appendix B's sweep finds survivors below $\theta^*(m)$ at every m). The finding is that it is self-referential. Because $\lambda_{\mathrm{hidden}}$ is rewritten by the memory itself, no fixed θ is admissible unconditionally, and (3) can carry an admissible step across the boundary within one pass. This places the published family. ATLAS trains because its write, though orthogonalised, is also magnitude-bounded, and, we argue, it is that bound, not the orthogonalisation, that closes the loop (Corollary 2); the raw Titans-style rule ships no bound, closest to the reports of §1.

Experiment 1: the loop, observed. We run the unbounded write rule, whose dynamics are exactly what Proposition 1 linearises. It diverges on every run above the step-size threshold, most within a single forward pass. Figure 2 shows why the failure is easy to miss. The prediction loss and the step gate stay flat while the memory's weights gain thirteen orders of magnitude (a): the world model normalises and gates the memory's output (an RMSNorm and a sigmoid gate) before it re-enters the network, so a diverging internal state never reaches the loss. Inside the fatal pass the three quantities of the loop (3), the gradient, the curvature and the weight norm, climb together (b), and the curvature tracks the hidden activation, as (2) predicts, not the weight norm. This is the self-referential loop caught in the act: each write grows the memory's own curvature, which grows the next write. Panel (c) places the realised step against the frozen-gate radius of Proposition 1. Most runs cross it before dying, but not all, so we read it as a scale check, not a per-run predictor.

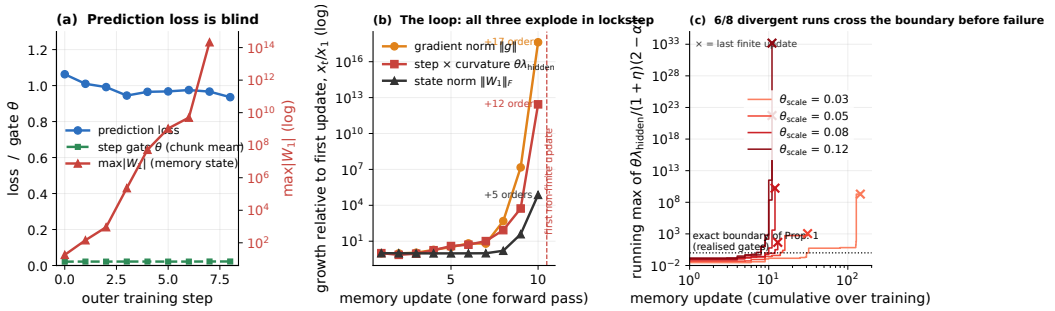

Figure 2: **The amplification loop, observed.** The unbounded update rule, $\theta_{\text{scale}} \in \{0.03, 0.05, 0.08, 0.12\}$, two seeds each. *(a)* Prediction loss and step gate flat while $\max|W_1|$ gains thirteen orders of magnitude ($\theta_{\text{scale}} = 0.03$; the run dies at outer step 8). *(b)* The fatal forward pass ($\theta_{\text{scale}} = 0.12$). $\|g\|$, $\theta\lambda_{\text{hidden}}$ and $\|W_1\|_F$ climb together, by $10^{17.6}$, $10^{12.4}$ and $10^{4.9}$ over the pass. The curvature tracks $\|h\|$ ($10^{6.3}$), not $\|W_1\|_F$, as (2) predicts. *(c)* Running maximum of $\theta\lambda_{\text{hidden}}/(1+\eta)(2-\alpha)$ at the realised gates. Six runs break through the boundary and die shortly after ($\times$: last finite update); two die *below* it ($\leq 0.51\times$).

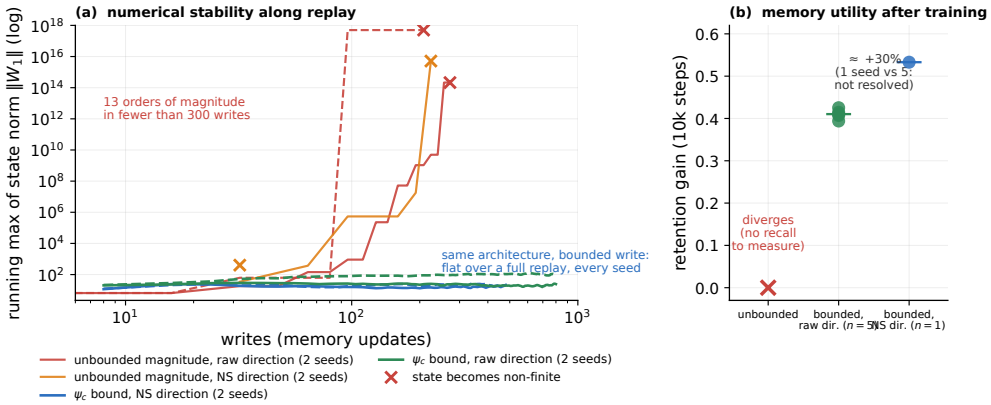

Figure 3: **Bounded magnitude is sufficient for stability without sacrificing memory utility.** *(a)* Running max of the state norm along replay. Unbounded magnitude ends non-finite with the raw direction (red, $\theta_{\text{scale}} = 0.03$) and the NS direction alike (orange, 0.08); every ψ_c-bounded run (blue/green, two seeds each) is flat. *(b)* Retention gain, individual seeds. The bounded raw-direction rule is stable and functional; the NS direction scores $\approx 30\%$ higher on one seed against five (not resolved); the unbounded rule leaves no recall to measure.

Experiment 2: the bound cuts the loop. Figure 3 tests sufficiency and utility together. Replayed over a stream of 800 writes after 10^4 training steps, both longest-lived unbounded runs end non-finite while every ψ_c-bounded run stays flat, on every seed (a). The converse control (orange) restores the raw magnitude on top of the Newton–Schulz direction. The threshold moves one rung (θ^* from ≈ 0.03 to 0.05–0.08), but the cliff remains. At $\theta_{\text{scale}} \geq 0.08$, 10 of 10 runs die within seven steps. Orthogonalisation alone does not stabilise the unbounded rule. The bound also leaves the memory functional (b). It stays stable and retentive with the raw direction (0.410 ± 0.011 over five seeds), and 0.53 with the Newton–Schulz direction on the same bound. That last arm has one scored seed against the raw direction's five, so the $\approx 30\%$ between them is unresolved and we draw no conclusion from it. What the panel establishes is that a bounded rule stays retentive at all, where the unbounded rule leaves no recall to measure. The stability comes from the magnitude bound, in either direction.

5 Second Failure Mode: Silent Collapse

A stable memory can still fail silently by losing its diversity. This collapse is learned during training, carried by a different learned gate on each platform, and baked into the weights. No replay-time fix restores it.

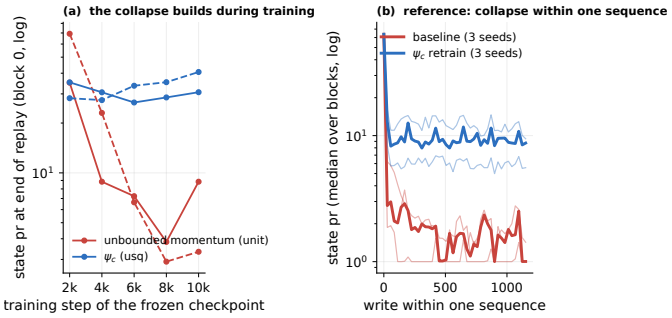

Figure 4: **The collapse phenotype, on both platforms.** *(a)* Testbed, state participation at the end of an identical replay, per training checkpoint (block 0, two seeds per arm). The collapsing arm loses its dimensions as outer training proceeds; ψ_c keeps them. *(b)* Community implementation, state participation along one sequence (thin: per-seed medians over 12 blocks, bold: median of three seeds). The trained baseline collapses from 63.7 to 1–3; the retrain under the magnitude coupling does not.

The *phenotype* is a loss of diversity. The writes, and then the state, concentrate in a nearly one-dimensional subspace. We measure it by the participation ratio (Gao et al., 2017) of a matrix's singular values, $\mathrm{pr}(W) = (\sum_i \sigma_i^2)^2 / \sum_i \sigma_i^4$, the number of directions that carry its energy. This is a different functional from the entropic effective rank of Roy & Vetterli (2007), and a low pr certifies no loss of rank ($\mathrm{pr}(\mathrm{diag}(1, \varepsilon)) \to 1$ while the matrix stays full-rank), so it flags a loss of well-conditioning, not of stored content. We compute it separately for the memory *state* (written pr) and for individual *writes* (r_{eff}, the same participation ratio averaged along the stream, not a matrix rank), reporting the two as distinct. In a collapsed memory the accumulated state concentrates in nearly one direction, so successive writes must largely align, and prediction loss barely reacts. On both platforms the collapse is learned by the outer loop and travels through the magnitude and diversity of the writes; the learned channel differs, so we present each in turn. A state whose energy spreads over many directions can separate many items; a collapsed one cannot. So we take diversity as the property worth protecting, and study what protects it.

The phenotype. On the testbed the collapse is acquired during outer training. Replaying frozen checkpoints identically, the state participation of the collapsing arm (trained without the magnitude coupling ψ_c) falls from 69 to 3 on one seed and from 35 to 9 on the other between 2,000 and 10,000 steps, while the ψ_c arm keeps its dimensions on both ($35 \to 31$, $28 \to 41$; Figure 4a). It is deterministic per seed, and severity tracks the learned diversity of the write stream. Across ten frozen blocks (five seeds, two layers), the only write stream that kept its diversity ($r_{\mathrm{eff}} = 55$) belongs to the only state that ends healthy ($\mathrm{pr} = 49.5$). Longer training does not repair it. At 30,000 steps the uncoupled arm is collapsed from step 4,000 on both seeds, while ψ_c holds with a slow late erosion ($\mathrm{pr} \approx 30 \to 13$), not an asymptotic fix.

On the reference implementation the same phenotype runs its course within a single sequence. From 63.7 at initialisation, the trained baseline's state participation falls to 1–3 on 12 of 12 measured blocks, three seeds (Figure 4b), through a different channel shown below.

Theory: a structural rank bound.

Remark 1 (structural rank bound). Each write is one chunk's gradient sum, m outer products, so a momentum-free write has rank at most m. A momentum write blends $\sim 1/(1-\eta)$ chunks, and only that blending can make it rich.

In practice the momentum-free rank sits near one, so a memory concentrates when its writes are few, individually poor, or mutually correlated. That correlation is itself a training-time lever: severing the read→write dependence raises write and state diversity by an order of magnitude, at ∼20% retention cost. The next experiment shows which learned gate controls it.

Experiment: which gate carries it (testbed). To find the cause we built *chimeras*: a frozen collapsed memory with one component swapped in from its healthy twin at replay time (Figure 5a). Only one swap brings write diversity back, the momentum head, on all three seed pairs (r_{eff} from 1.5–3.6 up to 60–68); the update rule, the projections, the initialisation and the other gates do noth-

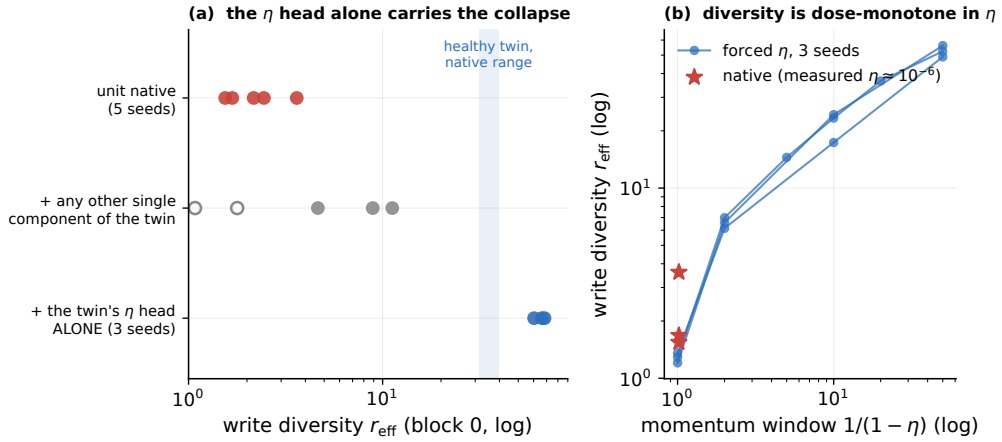

Figure 5: **Causal interventions on frozen twins (testbed, replay time).** *(a)* One healthy-twin component transplanted into the collapsed one. No other single component restores write diversity (open markers: writes disengaged); the momentum head alone reproduces the full rescue, 3/3 seed pairs. Shaded: the healthy twin's native range. *(b)* Momentum forced to a constant. Write diversity is monotone in the window $1/(1-\eta)$, $\times 40$ over the dose, three seeds; the native collapsed arm realises $\eta \approx 10^{-6}$ (stars).

ing. A direct reading agrees: in the collapsed memory the momentum head sits near zero (10^{-6} against the healthy twin's 0.99), so training has learned to switch momentum off. The effect is graded, not a threshold, rising smoothly as we force momentum from 0 to 0.98 (Figure 5b).

The carrier is platform-specific, so we flag the next result on its own. On the community implementation, pinning momentum for all of training does not stop the collapse; forgetting takes over instead ($\alpha_{\text{med}} = 0.88$), and the writes stay near rank one ($r_{\text{eff}} = 1.6$). Blocked on one gate, the loop reaches the same corner through the next. This rests on a single seed per arm, below our usual three to five, so we report it as an observation, not a demonstrated reroute.

Geometry is restorable; function is not. Bringing the geometry back does not bring the memory back. The same swaps that restore write diversity turn the memory's contribution at gap 0 from clearly positive ($+0.18$ and $+0.25$; its healthy twins reach $+0.31$ and $+0.38$) to clearly negative (-0.72 and -0.87 with momentum forced to a constant, -0.71 and -0.84 under the component transplant), and the repaired memory reads no content (paired gain zero, against $+0.38$ to $+0.53$ natively); matching write sizes rules out a scale artefact. The rest of the network has co-adapted to the collapsed, momentum-free writes, so a late repair only breaks what the network now expects. We therefore compare a memory's contribution only within its own arm, never across arms, since a collapsed memory can look fine merely by not interfering. Collapse is baked into the co-adapted weights during training, and no replay-time fix we tried undoes it. This is also why we rank no memory by quality. No instrument of ours reads content independently of the readout it co-adapted with.

The same collapse, on the community implementation. The phenotype reproduces on the most-used open-source implementation, through a different learned channel, not momentum but near-amnesic forgetting, with near-rank-one writes and a state participation that falls to 1–3 (Figure 4b). At the training horizon an extinct memory reads the same as no memory. The prediction loss does not separate the two (Appendix E). The full evidence, learning-rate robustness and the write-normalisation ablation, is in Appendix E.

Summary. Two learned routes, vanishing momentum on the testbed and near-amnesic forgetting on the community implementation, reach the same collapsed memory, and neither shows at the training horizon. Only the magnitude-constrained arm keeps a diverse memory. ψ_c is a control that shows collapse is avoidable, not a guaranteed fix.

The memory is used. On a synthetic associative-recall probe the memory beats the memory-free control at matched cost (accuracy ≈ 0.57 vs. 0.46 at $k = 8$, ≈ 0.35 vs. 0.26 at $k = 16$; three

seeds, Appendix F). On a task built to need memory, the module earns its keep. The two arms are indistinguishable here.

What low rank costs. A collapsed memory answers distinct queries with nearly the same vector. This is the regime Nazari & Rusch (2026) analyse for linear attention: when the read map has low effective rank, changing the query can change the output only a little. Their proof uses only the linear read, so it carries to our deep read through its Jacobian. The bound holds on every sampled query of our trained memories, and the collapsed arm has the lower read-map effective rank (Appendix D). This rank is not free. In a controlled probe we cap the read map toward rank one and measure recall. Recall at high load falls with the cap, from 0.47 at rank 4 to about 0.19 at rank 1 ($k = 16$; Appendix F). The geometry that collapse removes is the same geometry a memory needs to hold many items apart. This is the mechanism of the cost. It is not yet the cost of the learned collapse: the cap is imposed, not learned, and the rank-one run is hard to train on its own. On the trained arms the collapsed and diverse memories still score the same. Whether the learned collapse costs accuracy on a task that needs the memory, at larger scale, is the open question this work leaves.

6 Limitations and conclusion

Limitations. The evidence base is small by design. It rests on a custom testbed where causal tests are cheap and an external implementation where they reproduce, with 3–5 seeds per causal claim and 0.5M- and 20M-parameter models. The theory is elementary. Its Gauss–Newton diagnostic does not certify the full nonlinear, time-varying update, and its chunk analysis is measured only to $m \leq 32$ against deployed sizes up to ≥ 2048 (Appendix B). The design criterion assumes a forgetting floor and an exact write bound. Our trained arms learn a small forgetting rate with no enforced floor, and bound the write only up to the Newton–Schulz overshoot. They approximate the criterion rather than enforce it. Our budgets are short. On the testbed's prediction loss, the memory arm is worse than its memory-free ablation at the training horizon. On associative recall it does pay (Appendix F), so the loop is not switching off a useless module. But we cannot yet show that a healthy memory's advantage reaches long contexts. Separating a pathology from a rational bypass cleanly is the natural next step. It needs a task where the memory measurably pays at the training horizon, and at larger scale.

Conclusion. Test-time memories are attractive for one reason. Their cost stays fixed as the context grows, while an attention cache grows with it. That promise holds only while the memory stays healthy. The prediction loss does not track that health. It stays flat while the state builds toward divergence. It stays flat again while the memory collapses onto a single direction. Both failures act through the writes. Divergence acts through their magnitude. Collapse acts through their diversity.

Three take-aways carry over to anyone building such a memory. First, constrain the write. A per-step magnitude bound, with the memory timescale, removes the divergence at its source, where orthogonalizing the direction alone does not. Second, watch the rank, not the loss. A memory can hold a low, stable loss while its effective rank falls to one, so track the rank of the write it accumulates or of its read map. Third, measure retention directly. A latent episodic probe reads what the memory keeps at a chosen distance, free of any decoder, and with the write geometry separates a diverse memory from a collapsed one where accuracy alone does not. These lessons matter most for long-term memory, for long documents, agent sessions and hours of video, where a healthy-looking loss can hide a memory decayed to a single pattern. Report stability and retention beside it.

Reproducibility statement

The theoretical analysis is supported by an anonymized repository, `https://github.com/255212321/titans-theory`, containing the reference deep test-time memory implementation studied here (a Titans-style deep memory with the ATLAS Omega-rule sliding window) together with self-contained first-principles simulations of its write-diversity and memory-span behaviour; each figure the repository produces is regenerated by a single script (see its README). The full test-time-training testbed (the numerical-divergence and rank-collapse measurements of the paper) is specified in the appendix, and its code and cached latents will be released publicly upon acceptance of the paper.

The Use of Large Language Models

We used LLM-based coding and writing assistants throughout this project, and we take full responsibility for the final content of this paper, including all text, mathematics, claims, code, and other artifacts produced with their assistance. The assistants were also used for coding support, including implementing, running, and debugging the experiment and analysis code and generating plots, and for routine research support such as checking derivations, numerically verifying mathematical statements, and locating related work. The authors reviewed and validated all AI-assisted code, text, and mathematics, and take full responsibility for the final content of the paper.

References

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. In *International Conference on Machine Learning (ICML)*, 2022.

Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on edge of stability in deep learning. In *International Conference on Machine Learning (ICML)*, 2022.

Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Ré. Zoology: Measuring and improving recall in efficient language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Ali Behrouz, Zeman Li, Praneeth Kacham, Majid Daliri, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. ATLAS: Learning to optimally memorize the context at test time. *arXiv preprint arXiv:2505.23735*, 2025a.

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b.

Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. It's all connected: A journey through test-time memorization, attentional bias, retention, and online optimization. In *International Conference on Learning Representations (ICLR)*, 2026.

Yingfa Chen, Xinrong Zhang, Shengding Hu, Xu Han, Zhiyuan Liu, and Maosong Sun. Stuffed Mamba: Oversized states lead to the inability to forget. In *Conference on Language Modeling (COLM)*, 2025.

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 *International Conference on Learning Representations (ICLR)*, 2021.

Alex Damian, Eshaan Nichani, and Jason D. Lee. Self-stabilization: The implicit bias of gradient descent at the edge of stability. In *International Conference on Learning Representations (ICLR)*, 2023.

Gavriel Di Nepi, Federico Siciliano, and Fabrizio Silvestri. Titans revisited: A lightweight reimplementation and critical analysis of a test-time memory model. *arXiv preprint arXiv:2510.09551*, 2025.

Shibhansh Dohare, J. Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A. Rupam Mahmood, and Richard S. Sutton. Loss of plasticity in deep continual learning. *Nature*, 632:768–774, 2024.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning (ICML)*, 2021.

Peiran Gao, Eric Trautmann, Byron M Yu, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, 2017.

Alex Graves, Greg Wayne, and Ivo Danihelka. Neural Turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538 (7626):471–476, 2016.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

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

Eliahu I. Jury. *Theory and Application of the z-Transform Method*. Wiley, 1964.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning (ICML)*, 2020.

Aviral Kumar, Rishabh Agarwal, Dibya Ghosh, and Sergey Levine. Implicit under-parameterization inhibits data-efficient deep reinforcement learning. In *International Conference on Learning Representations*, 2021.

Zeman Li, Ali Behrouz, Yuan Deng, Peilin Zhong, Praneeth Kacham, Mahdi Karami, Meisam Razaviyayn, and Vahab Mirrokni. TNT: Improving chunkwise training for test-time memorization. In *International Conference on Learning Representations (ICLR)*, 2026.

Bo Liu, Rui Wang, Lemeng Wu, Yihao Feng, Peter Stone, and Qiang Liu. Longhorn: State space models are amortized online learners. *arXiv preprint arXiv:2407.14207*, 2024.

Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In *International Conference on Machine Learning*, 2023.

Luke Metz, Niru Maheswaranathan, Jeremy Nixon, C. Daniel Freeman, and Jascha Sohl-Dickstein. Understanding and correcting pathologies in the training of learned optimizers. In *International Conference on Machine Learning (ICML)*, 2019.

Philipp Nazari and T. Konstantin Rusch. The key to state reduction in linear attention: A rank-based perspective. *arXiv preprint arXiv:2602.04852v2*, 2026.

Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *International Conference on Machine Learning (ICML)*, 2023.

Young-Jae Park, Minseok Seo, and Hae-Gon Jeon. VideoTitans: Scalable video prediction with integrated short- and long-term memory. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning (ICML)*, 2013.

Boris T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *European Signal Processing Conference (EUSIPCO)*, 2007.

S. Roy and J. J. Shynk. Analysis of the momentum LMS algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(12):2088–2098, 1990.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In *International Conference on Machine Learning (ICML)*, 2021.

Jürgen Schmidhuber. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 4(1):131–139, 1992.

Julien Siems, Timur Carstensen, Arber Zela, Frank Hutter, Massimiliano Pontil, and Riccardo Grazzi. DeltaProduct: Improving state-tracking in linear RNNs via householder products. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning (ICML)*, 2020.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): RNNs with expressive hidden states. *arXiv preprint arXiv:2407.04620*, 2024.

Arnuv Tandon, Karan Dalal, Xinhao Li, Daniel Koceja, Marcel Rød, Sam Buchanan, Xiaolong Wang, Jure Leskovec, Sanmi Koyejo, Tatsunori Hashimoto, Carlos Guestrin, Jed McCaleb, Yejin Choi, and Yu Sun. End-to-end test-time training for long context. *arXiv preprint arXiv:2512.23675*, 2025.

Bohao Tang, Zhen Qin, Yuqi Pan, Zheng Li, Pengfei Liu, and Ya Zhang. Modular TTT: Rethinking test-time training as composable modules. *arXiv preprint arXiv:2608.07110*, 2026.

Shohei Taniguchi et al. ADOPT: Modified Adam can converge with any β_2 with the optimal rate. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Neehal Tumma, Noel Loo, and Daniela Rus. Preconditioned DeltaNet: Curvature-aware sequence modeling for linear recurrences. *arXiv preprint arXiv:2604.21100*, 2026.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning (ICML)*, 2023.

Johannes von Oswald, Nino Scherrer, Seijin Kobayashi, Luca Versari, Songlin Yang, Sarthak Mittal, Maximilian Schlegel, Kaitlin Maile, Yanick Schimpf, Oliver Sieberling, Alexander Meulemans, Rif A. Saurous, Guillaume Lajoie, Charlotte Frenkel, Razvan Pascanu, Blaise Agüera y Arcas, and João Sacramento. MesaNet: Sequence modeling by locally optimal test-time training. In *International Conference on Learning Representations (ICLR)*, 2026.

Ke Alexander Wang, Jiaxin Shi, and Emily B. Fox. Test-time regression: A unifying framework for designing sequence models with associative memory. *arXiv preprint arXiv:2501.12352*, 2025.

Phil Wang. titans-pytorch. `https://github.com/lucidrains/titans-pytorch`, 2025.

Bernard Widrow and Marcian E. Hoff. Adaptive switching circuits. In *IRE WESCON Convention Record*, pp. 96–104, 1960.

Jingkang Yang, Shuai Liu, Hongming Guo, Yuhao Dong, et al. EgoLife: Towards egocentric life assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving Mamba2 with delta rule. In *International Conference on Learning Representations (ICLR)*, 2025b.

Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations (ICLR)*, 2020.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T. Freeman, and Hao Tan. Test-time training done right. In *International Conference on Learning Representations (ICLR)*, 2026.

Code, the exact configuration of every arm, and the time-stamped pre-registration ledger (predictions with probabilities written down before each launch, and the recorded verdicts, including the negative ones) accompany the submission as supplementary material.

A PROOFS

The lemmas and corollaries proved here are stated where they are used, in Appendices B and C; each proof names its target.

Proof of Proposition 1. Along the eigendirection of H with eigenvalue λ, write (δ_t, s_t) for the deviations of state and momentum from their fixed point. Linearised around the fixed point, with the step absorbed into the momentum variable, the recurrence reads $s_t = \eta\, s_{t-1} - \theta\lambda\, \delta_{t-1}$ and $\delta_t = (1-\alpha)\, \delta_{t-1} + s_t$; substituting the first into the second gives

$$\begin{pmatrix} \delta_t \\ s_t \end{pmatrix} = A(\lambda) \begin{pmatrix} \delta_{t-1} \\ s_{t-1} \end{pmatrix}, \qquad A(\lambda) = \begin{pmatrix} (1-\alpha) - \theta\lambda & \eta \\ -\theta\lambda & \eta \end{pmatrix},$$

with trace $T = (1-\alpha) + \eta - \theta\lambda$ and determinant $D = \eta(1-\alpha)$. Both roots of $z^2 - Tz + D$ lie in the open unit disk if and only if the Jury conditions (Jury, 1964) hold: $|D| < 1$ and $|T| < 1 + D$. The first is immediate, since $\eta(1-\alpha) \in [0,1)$ for $\eta \in [0,1)$, $\alpha \in [0,1]$. The second splits into two inequalities: $T < 1 + D$ reduces to $\theta\lambda > -\alpha(1-\eta)$, trivially true for $\theta\lambda \geq 0$ except when both $\alpha = 0$ and $\lambda = 0$, where equality is marginal; and $T > -(1+D)$ reduces to $\theta\lambda < (1-\alpha) + \eta + 1 + \eta(1-\alpha) = (1+\eta)(2-\alpha)$. □

Remark 2 (scope of the frozen-gate condition). The exact region of Proposition 1 is wider than the classical threshold exactly when $\eta > \alpha/(2-\alpha)$. Momentum enlarges the admissible step, the heavy-ball effect (Polyak, 1964), while forgetting alone narrows it; the condition holds on the bounded arms we run and fails on the collapsed ones on the testbed, whose momentum gate reaches 10^{-6} (§5). With momentum all but extinguished, $\eta < \alpha/(2-\alpha)$ for any appreciable forgetting, so that the region there is *narrower* than the classical one. And for frozen gates and fixed curvature, momentum and forgetting alone do not predict instability below this boundary. The real iteration is neither frozen nor fixed. The learned gates vary along the stream, making the linearised dynamics a product of time-varying matrices, for which step-wise spectral stability is neither necessary nor sufficient. Individually stable but non-normal factors can amplify transiently, and a switching sequence can chain those transients. Throughout the paper, Proposition 1 is therefore used as a diagnostic to be tested experimentally, not as a certificate.

Proof of Lemma 1. By the triangle inequality, $\|W_t\|_F \leq (1-\alpha_t)\|W_{t-1}\|_F + b \leq (1-\alpha)\|W_{t-1}\|_F + b$; unrolling gives $\|W_t\|_F \leq (1-\alpha)^t \|W_0\|_F + b\sum_{s=0}^{t-1}(1-\alpha)^s = (1-\alpha)^t\|W_0\|_F + \frac{b}{\alpha}(1-(1-\alpha)^t)$, a convex combination of $\|W_0\|_F$ and b/α, hence at most their maximum. For attainment take $U_t \equiv b\,V$ with $\|V\|_F = 1$ and $\alpha_t \equiv \alpha$: the recurrence is scalar along V and converges to $(b/\alpha)V$. □

Proof of Lemma 2. With $\alpha_t \equiv \alpha$ the recurrence unrolls to the *linear* combination $W_t = (1-\alpha)^t W_0 + \sum_{s=1}^{t}(1-\alpha)^{t-s}U_s$ with deterministic coefficients, so pairwise uncorrelatedness kills every cross term in $\mathbb{E}\|W_t\|_F^2$:

$$\mathbb{E}\|W_t\|_F^2 = (1-\alpha)^{2t}\|W_0\|_F^2 + b^2\sum_{s=1}^{t}(1-\alpha)^{2(t-s)} \xrightarrow[t\to\infty]{} \frac{b^2}{1-(1-\alpha)^2} = \frac{b^2}{\alpha(2-\alpha)}.$$

□

Proof of Corollary 2. Lemma 1 applied to each fast tensor gives $\|W_{i,t}\|_F \le B$ for $i \in \{1,2\}$ and all t. For the readout block, $h = \sigma(W_1 k)$ with $\sigma(0) = 0$ and σ L-Lipschitz coordinatewise gives $\|h\| \le L\|W_1 k\| \le L\,\|W_1\|_F\,R \le LRB$, so $\lambda_{\max}(H^{(W_2)}) = 2\|h\|^2 \le 2L^2R^2B^2$. For the hidden block, $\lambda_{\max}(H^{(W_1)}) \le 2\|k\|^2\|W_2\,\mathrm{diag}\,\sigma'\|_2^2 \le 2R^2L^2\|W_2\|_F^2 \le 2L^2R^2B^2$. The coupled update moves both tensors at once, so the joint Gauss–Newton proxy is $H_{\mathrm{GN}} = 2J^\top J$ with $J = [\,J_1 \;\; J_2\,]$ the Jacobian of the residual. Since $JJ^\top = J_1J_1^\top + J_2J_2^\top$ and $J_2J_2^\top = \|h\|^2 I$ is isotropic, its top eigenvalue adds to that of $J_1J_1^\top$, so $\lambda_{\max}(H_{\mathrm{GN}}) = \lambda_{\max}(H^{(W_1)}) + \lambda_{\max}(H^{(W_2)}) \le 4L^2R^2B^2$. Any $\theta < (1+\eta)(2-\alpha)/(4L^2R^2B^2)$ therefore satisfies $\theta\lambda < (1+\eta)(2-\alpha)$ for the joint curvature, which is the upper stability inequality of the frozen quadratic surrogate in Proposition 1. Positive forgetting supplies its lower inequality. This argument bounds the proxy, not the residual Hessian or the time-varying nonlinear transition. □

Proof of Proposition 2. Within a chunk the gradient is evaluated at the anchor, so the momentum accumulates against the *fixed* entering error δ_0: for $j = 1, \dots, m$,

$$s_j \;=\; \eta\, s_{j-1} - \theta\lambda\,\delta_0 \;=\; \eta^j s_0 \;-\; \theta\lambda\,\frac{1-\eta^j}{1-\eta}\,\delta_0,$$

while the state accumulates $\delta_j = a\,\delta_{j-1} + s_j$. Unrolling the latter over the chunk,

$$\delta_m \;=\; a^m\delta_0 + \sum_{j=1}^{m} a^{m-j}s_j \;=\; \Big(a^m - \theta\lambda\,\frac{1}{1-\eta}\sum_{j=1}^{m} a^{m-j}\big(1-\eta^j\big)\Big)\,\delta_0 \;+\; \Big(\sum_{j=1}^{m} a^{m-j}\,\eta^j\Big)\,s_0,$$

which is the first row of $B(\lambda)$; the second row is the $j = m$ case of the momentum identity above. The chunked dynamics is the iteration of the fixed matrix $B(\lambda)$ on chunk boundaries, hence locally stable if and only if $\rho(B(\lambda)) < 1$ for every eigenvalue λ of the curvature. For the asymptotics of κ: at $\alpha \to 0$ ($a = 1$) the sum gives $\kappa = \big(m - \eta\,\frac{1-\eta^m}{1-\eta}\big)/(1-\eta)$, so $\kappa \sim m/(1-\eta)$ as $m \to \infty$ at fixed η. □

Proof of Corollary 1. By definition κ/m is the arithmetic mean of the terms $c_j = a^{m-j}(1-\eta^j)/(1-\eta)$, $j = 1, \dots, m$. Each factor satisfies $a^{m-j} \le 1$ and $(1-\eta^j)/(1-\eta) \le 1/(1-\eta)$, hence $\kappa/m \le 1/(1-\eta)$; at $m = 1$ the single term is $(1-\eta)/(1-\eta) = 1$. At $\alpha = 0$ the terms $c_j = (1-\eta^j)/(1-\eta)$ form an increasing sequence in j, and the mean of the first m terms of an increasing sequence is nondecreasing in m. The individual terms converge geometrically, $1/(1-\eta) - c_j = \eta^j/(1-\eta)$, but their mean does not:

$$\frac{1}{1-\eta} - \frac{\kappa}{m} \;=\; \frac{\eta\,(1-\eta^m)}{m\,(1-\eta)^2},$$

which decays as $\Theta(1/m)$ for fixed $0 < \eta < 1$; it is zero when $\eta = 0$. At $m \approx 1/(1-\eta)$ the mean has reached only $\approx 37\%$ of the cap, so the approach is slow where η is large: at $\eta = 0.97$, $m = 32$, $\kappa/m = 12.4$ against a cap of 33.3. □

B THE CHUNK FACTOR

With chunk length m, gradients are evaluated at the state frozen at the chunk's start (§4). The surrogate below assumes fixed gates, a common quadratic curvature and one anchored gradient per token. Under these assumptions, κ describes the accumulated effect of the entering error and κ/m its per-token average. These expressions depend on the update convention: applying a summed gradient once per chunk gives a different map. The surrogate does not model the heterogeneous keys, residual curvature or adaptive gates of the complete testbed.

Proposition 2 (chunk transfer matrix and effective step)**.** *Under the assumptions of Proposition 1, with a common per-token curvature λ, let the m gradients of a chunk all be evaluated at the state frozen at the chunk's start, and write $a = 1 - \alpha$. The chunk-to-chunk map in the eigendirection is $(\delta_m, s_m)^\top = B(\lambda)\,(\delta_0, s_0)^\top$ with*

$$B(\lambda) = \begin{pmatrix} a^m - \theta\lambda\,\kappa & P_m \\ -\theta\lambda\,\dfrac{1-\eta^m}{1-\eta} & \eta^m \end{pmatrix}, \qquad \kappa = \frac{1}{1-\eta}\sum_{j=1}^{m} a^{m-j}\big(1-\eta^j\big), \quad P_m = \sum_{j=1}^{m} a^{m-j}\,\eta^j,$$

and the chunked recurrence is locally stable if and only if $\rho(B(\lambda)) < 1$ for every eigenvalue λ of the curvature.

Over one chunk the entering error is acted on with accumulated weight $\theta\kappa$. A chunk also processes m tokens, so the natural unit is the per-token action $\theta\,\kappa/m$.

Corollary 1 (the per-token action is capped by the momentum factor)**.** *Under the assumptions of Proposition 2, $\kappa(m,\eta,\alpha)/m$ is the arithmetic mean of the m momentum load factors $a^{m-j}(1-\eta^j)/(1-\eta)$, $j = 1,\dots,m$. Consequently*

$$\kappa(1,\eta,\alpha) = 1, \qquad \frac{\kappa(m,\eta,\alpha)}{m} \;\le\; \frac{1}{1-\eta} \qquad (m \ge 1),$$

and at $\alpha = 0$ the ratio κ/m is nondecreasing in m and approaches $1/(1-\eta)$ with a gap of $\Theta(1/m)$ for fixed $0 < \eta < 1$ (the gap is zero at $\eta = 0$). In the high-momentum regime, at $m \approx 1/(1-\eta)$ it has reached only $\approx 37\%$ of the cap, and $m \gtrsim 10/(1-\eta)$ is needed for 90%. Chunked processing therefore multiplies the per-token step by at most the momentum factor $1/(1-\eta)$, uniformly in the chunk length; within the grid we measure ($m \le 32$) the cap is reached to 73% at $\eta = 0.9$, 37% at $\eta = 0.97$, and 15% at the $\eta \approx 0.99$ our arms realise.

Measurement protocol and result. On the testbed's witness runs we measure the divergence threshold θ^* at initialisation over the grid

$$m \in \{1, 2, 4, 8, 16, 32\}, \qquad \eta_{\text{init}} \in \{0.05, 0.3, 0.6, 0.9, 0.97\}.$$

We use 18 step scales (a geometric ladder from 0.013 to 2.6) and two seeds per cell, giving 1080 short runs with the unbounded rule. A run is classified as divergent if its state becomes non-finite within three outer steps; this is a fast-blow-up test, not a long-horizon stability boundary, so θ^* marks the onset of immediate divergence rather than of eventual slow growth. θ^* is the geometric mean of the last surviving and first dying rung; κ is evaluated at the median realised gates of the surviving runs. What the corollary proves and what the grid measures do not coincide, so we keep them apart (Figure 6).

Measured: θ^* falls with m up to $m \approx 8$ (at $\eta = 0.05$, $2.2 \to 0.64$), then is flat to $m = 32$ at every η. At $m = 32$ the product $\theta^*\kappa/m$ is constant across η to $\times 1.5$ ($\times 3$ at $m = 8$) while κ/m spans 1.0 to 12.1, so this rescaling reduces the variation across momentum settings. It does not derive the threshold from Corollary 1.

Proved: only that saturation. The exact boundary $\rho(B(\lambda)) < 1$ keeps falling, and at the momentum our arms realise it falls far faster than $1/m$. Which Jury condition binds switches with η. At $\eta = 0.05$ the step condition binds and the boundary falls $\times 35$ from $m = 1$ to 32, close to $1/m$. From $\eta \approx 0.9$ upward it is $|\det B| < 1$ that binds, and since (at $\alpha = 0$)

$$\det B = \eta^m + \theta\lambda\,C_m, \qquad C_m = \frac{\eta\,(1-\eta^m)}{(1-\eta)^2} - \frac{m\,\eta^m}{1-\eta},$$

the boundary is $\theta\lambda < (1-\eta^m)/C_m$ in closed form. It reproduces the numerical $\rho(B) = 1$ to six figures wherever the determinant binds ($\eta \gtrsim 0.9$); at $\eta = 0.05$, where the step condition binds instead, it overstates the true boundary by $\times 20$ to $\times 300$. The mechanism is visible in the determinant: the unanchored system has $\det(A)^m = \eta^m$, so anchoring injects an area expansion proportional to $\theta\lambda$, and momentum widens the region at $m = 1$ but narrows it from $m \ge 2$. The narrowing is linear in $(1-\eta)$: as $\eta \to 1$, $C_m \to m(m-1)/2$ and the boundary tends to $2(1-\eta)/(m-1)$ (at $m = 32$, $\eta = 0.9999$ the closed form gives 6.455×10^{-6} against 6.452×10^{-6}), so the $m = 1$ to $m = 32$ ratio grows by a decade for each decade of $(1-\eta)$. That is a different dependence from the $1/(1-\eta)$ cap of Corollary 1, which bounds a different quantity.

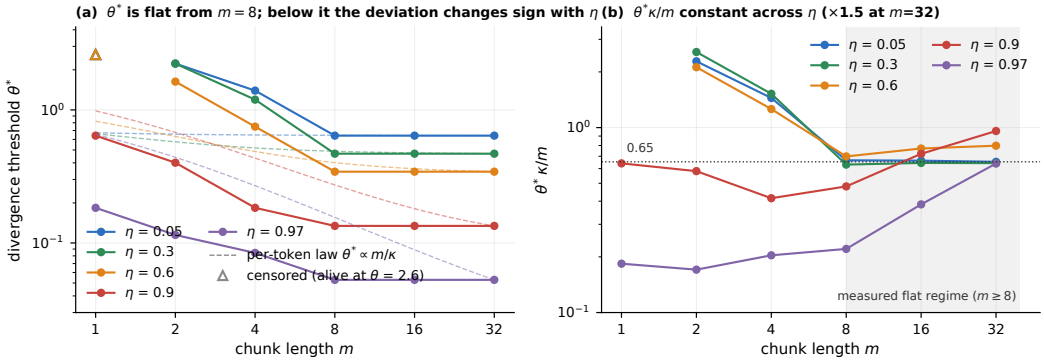

Figure 6: **The measured threshold plateaus in the chunk length.** Divergence threshold θ^* at initialisation, grid $m \times \eta$ (two seeds per cell; κ at realised gates). *(a)* θ^* flat from $m = 8$; dashed: the per-token rescaling anchored at $m = 32$; small chunks sit above it at $\eta \leq 0.6$ and below it at $\eta \geq 0.9$; triangles: censored cells. *(b)* Across η at $m = 32$, $\theta^*\kappa/m$ is constant to $\times 1.5$ ($\times 3$ at $m = 8$) while κ/m spans 1.0–12.1 at the realised gates (against 12.4 at the init gate of Corollary 1; shaded: the measured flat regime; the exact boundary of Proposition 2 keeps falling over this range, and faster than $1/m$ at high η, see Appendix B).

Thus the frozen-anchor surrogate predicts a different chunk-size dependence from the short-run measurements beyond $m \approx 8$; we report the plateau as a measurement, not as a consequence of Corollary 1. Below $m = 8$ the deviation changes sign with η (the measured threshold sits above the rescaling for $\eta \leq 0.6$, below it for $\eta \geq 0.9$); we observe both and model neither. The arms we train realise $\eta \approx 0.99$, so it is that regime most relevant to these trained testbed arms.

C THE DESIGN PAIR (WRITE BOUND, FORGETTING)

Bounding each write alone does not ensure a uniform state bound. The state is a discounted sum of writes, and its limiting size is a property of the *pair* (write bound, forgetting). Two elementary lemmas frame that size for any rule whose write norm is bounded by a constant b: a worst-case upper bound, and the size decorrelated writes would produce. The second is a *reference scale*, not a floor. A bounded rule may write nothing at all, and anticorrelated writes fall below it (proofs in Appendix A).

Lemma 1 (worst case: the reachable ball)**.** *Let* $W_t = (1 - \alpha_t)\, W_{t-1} + U_t$ *with* $\|U_t\|_F \leq b$ *and* $0 < \alpha \leq \alpha_t \leq 1$. *Then for every* t

$$\|W_t\|_F \;\leq\; (1-\alpha)^t \|W_0\|_F \;+\; \frac{b}{\alpha}\Big(1 - (1-\alpha)^t\Big) \;\leq\; \max\big\{\|W_0\|_F,\; b/\alpha\big\},$$

and the asymptotic bound b/α *is attained by aligned writes* ($U_t \equiv b\,V$, $\alpha_t \equiv \alpha$).

Lemma 2 (typical size: decorrelated writes)**.** *If moreover* $\alpha_t \equiv \alpha$, $\mathbb{E}\|U_t\|_F^2 = b^2$, *and the writes are pairwise uncorrelated* ($\mathbb{E}\langle U_s, U_t\rangle = 0$ *for* $s \neq t$, $\mathbb{E}\langle W_0, U_t\rangle = 0$), *then*

$$\mathbb{E}\|W_t\|_F^2 \;\xrightarrow[t\to\infty]{}\; \frac{b^2}{\alpha\,(2-\alpha)}, \qquad \textit{an RMS size} \;\approx\; b/\sqrt{2\alpha}.$$

The two regimes differ by $\sqrt{(2-\alpha)/\alpha}$, an order of magnitude at the rates below. These are useful reference scales for examining write correlation (Figure 7), not a universal two-sided interval. Under the following assumptions, the state bound also bounds the Gauss–Newton curvature.

Corollary 2 (Gauss–Newton bound and surrogate margin)**.** *Let* $\mathcal{M}_W(k) = W_2\,\sigma(W_1 k)$ *with* σ *differentiable and L-Lipschitz,* $\sigma(0) = 0$, *keys* $\|k_t\| \leq R$, *and let both fast tensors follow* (1) *with* $\|U_t\|_F \leq b$ *and gates in* $\alpha_t \in [\alpha_-, \alpha_+] \subset (0, 1]$, $0 \leq \eta_- \leq \eta_t < 1$. *Write* $B = \max\{\|W_{1,0}\|_F, \|W_{2,0}\|_F, b/\alpha_-\}$. *Then along the trajectory each block curvature* (2) *satisfies* $\lambda \leq 2L^2R^2B^2$, *and the joint Gauss–Newton curvature of the coupled* (W_1, W_2) *update satisfies*

$$\lambda_{\max}(H_{\mathrm{GN}}) = \lambda_{\max}\big(H^{(W_1)}\big) + \lambda_{\max}\big(H^{(W_2)}\big) \leq 4\,L^2R^2B^2.$$

For $B > 0$, any fixed positive step

$$\theta \;<\; \frac{(1+\eta_-)(2-\alpha_+)}{4\,L^2R^2B^2}$$

therefore satisfies the quadratic-surrogate inequalities of Proposition 1 when its Hessian is replaced by the current Gauss–Newton matrix. This bounds the step, not the memory. It is not a stability certificate for the nonlinear update or for a sequence of time-varying transitions, and boundedness of W_t implies neither contraction between nearby trajectories nor that any memory content is retained.

The two bounds on α play different roles: α_- bounds state accumulation, while α_+ sets the smallest frozen-step margin. The general restriction $\alpha_t \le 1$ already supplies a conservative choice $\alpha_+ = 1$; a tighter upper bound improves that margin.

The raw-write rule does not supply the assumptions of Lemma 1; this does not imply that all its trajectories are unbounded. The corollary controls a Gauss–Newton surrogate, as in Remark 2, and does not include the residual Hessian or the derivative of the nonlinear write map. On our testbed, σ is SiLU ($L \le 1.1$) and key normalisation gives $R = 1$.

Substituting extrema observed during replay gives a positive surrogate step margin in each magnitude-controlled arm. The resulting B is set by the learned initialisation rather than b/α_-. This is a finite-replay calculation, not a design certificate when gate caps or global forgetting bounds are absent. The majorant $4L^2R^2B^2$ exceeds the measured joint Gauss–Newton curvature by one to two orders of magnitude, and the learned steps can exceed the corresponding conservative margin. The observed minimum momentum falls to $\approx 10^{-10}$ on collapsed arms (below the median 10^{-6} of §5) and stays at 0.38–0.82 on the magnitude-controlled arms, further changing this margin.

For constant forgetting, the participation of the squared coefficients in the discounted sum is $\big(1+(1-\alpha)^2\big)/\big(\alpha(2-\alpha)\big)$, asymptotically $1/\alpha$ for small α. This characterises the decay weights, not the recoverable information of a nonlinear memory whose later writes depend on earlier states.

Experiment: comparing state-size reference scales. (Instantiation: for the clipped rule $b = c$ exactly, so the lemmas apply verbatim; for the Newton–Schulz rules $b \approx \theta_{\max}\sqrt{r}$ only when the step gate has a cap, true by construction on the community implementation, while on our testbed the gate is uncapped and the bound only empirical.) We measure the realised gates of the four trained magnitude-controlled arms of Figure 3 in-replay. The learned forgetting settles at $\bar\alpha \approx 1.5$–2.1×10^{-3} per token, i.e. $\alpha_C = 1-(1-\bar\alpha)^C \in [0.012, 0.017]$ per optimizer call ($C = 8$ tokens), with the write norm b measured on the same replay. Six further replays at forced $\alpha \in \{0.003, 0.01, 0.03\}$ give ten $(b, \alpha_C, \text{plateau})$ triples spanning $\times 8$ in b and $\times 20$ in α_C (Figure 7). All measured plateaus lie between the decorrelated reference scale and the aligned-write bound. The observed log-slope is -0.44, closer to $-1/2$ than to -1, and the measured scale is approximately linear in b. Forced-α replays lie near the decorrelated reference; trained arms lie about $\times 4$ above it, consistent with temporal write correlation. These observations do not make the decorrelated scale a lower bound, or establish that low per-write rank alone causes the larger state norms. The measured settling timescale is consistent with the transient of Lemma 1: $1/\alpha_C \approx$ 60–85 calls ($\sim$500–700 tokens).

Different gate regimes across platforms. The two settings exhibit different gate behaviour. On the testbed, the uncoupled arm has median $\eta = 10^{-6}$, compared with 0.99 for its comparison twin (§5). On the community implementation, $\alpha_{\text{med}} = 0.54$–$0.79$ per chunk rapidly attenuates the direct contribution of preceding states, while momentum remains moderate (0.32–0.52). These decay rates do not alone determine functional retention because later writes can depend on earlier state information.

For comparison, the four magnitude-controlled testbed arms in Figure 3 have $\bar\alpha \approx 1.5$–2.1×10^{-3} per token, corresponding to a nominal decay timescale of roughly 500–700 token updates. These values are measured on constrained arms, not on the testbed's uncoupled arm; the platforms also apply decay in different units. They should therefore not be read as a direct cross-platform comparison of learned memory span.

Design implications. An explicit write cap and a positive forgetting floor provide a bounded-state guarantee. Their values should also be evaluated against the desired retention horizon. Stronger forgetting can control state growth while shortening the direct persistence of past writes. The present

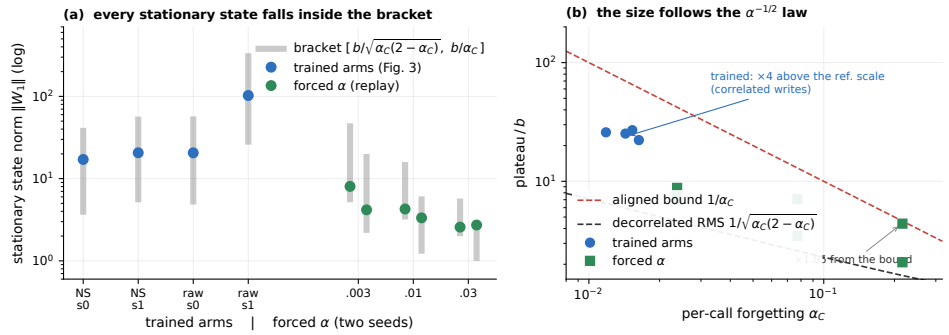

Figure 7: **Measured state sizes relative to two reference scales.** Ten replay measurements. Four trained arms (realised $\bar{\alpha}$, in-replay) and six forced-α replays. *(a)* Every stationary state lies between the aligned upper bound of Lemma 1 and the decorrelated reference of Lemma 2. *(b)* Plateau$/b$ against α_C. Forced-α points follow the $\alpha^{-1/2}$ law near the decorrelated reference scale; trained arms sit $\times 4$ above, consistent with greater temporal write correlation (§5). For trained arms the measured b conflates write and decay ($\pm 40\%$ at most; exact for forced α). The observed ordering relative to these scales is unchanged by this uncertainty.

results motivate these controls and associated diagnostics, but do not establish that fixed or constrained gates are universally better than adaptive ones.

D Reproducing Nazari's retrieval-error bound

Nazari & Rusch (2026) bound, for a *linear* read $o = Sq$, the amplification of a query perturbation by the effective rank of the read map. With $\tilde{q} = q^\star + n$, $R := (\|o - o^\star\|/\|o^\star\|)/(\|n\|/\|q^\star\|)$, stable rank $\mathrm{er}(S) := \|S\|_F^2/\|S\|_2^2$, and δ, γ the alignment of $n, q^\star$ with the top right singular vector of S, their bound is $\delta/\sqrt{\mathrm{er}(S)} \le R \le \sqrt{\mathrm{er}(S)}/\gamma$. The proof uses only the linear map, not the recurrence, so it transposes to our deep read $\mathcal{M}_W(q) = W_2\,\sigma(W_1 q)$ through its Jacobian $J(q) = W_2\,\mathrm{diag}(\sigma'(W_1 q))\,W_1$: to first order in $\|n\|$, $o - o^\star = J(q^\star)\,n$, and

$$\frac{\delta}{h(q^\star)\sqrt{\mathrm{er}(J)}} \;\le\; R \;\le\; \frac{\sqrt{\mathrm{er}(J)}}{h(q^\star)\,\gamma}, \qquad h(q) := \frac{\|\mathcal{M}_W(q)\|}{\|J(q)\,q\|}, \tag{4}$$

with δ, γ on the right singular vectors of $J(q^\star)$ and h the homogeneity defect ($h \equiv 1$ for ReLU without bias; measured per query for our SiLU/GELU).

Measurement. On frozen trained checkpoints of the collapsed (no-ψ_c) and healthy (ψ_c) arms (three seeds, 8 and 16 written pairs, 256 queries per block, isotropic/aligned/orthogonal perturbations at $\varepsilon \in \{10^{-3}, 10^{-1}\}$), we compute $\mathrm{er}(J)$, h and the empirical R, and test the bound.

Result. The alignment-aware bound holds on *every* sampled query, both arms, both loads, all seeds. The read-map effective rank is lower on the collapsed arm ($\mathrm{er}(J) = 1.13$ vs. 1.64 at 16 pairs), and its empirical noise amplification R is about $17\times$ larger, consistent with the bound's mechanism, measured on our own memory rather than imported. As the bound makes explicit, R also depends on the alignment terms and the homogeneity defect, so we read the rank as one contributing factor, not the sole cause.

Caveats. Three, stated plainly. (i) Only the alignment-aware bound is informative. The alignment-free form $1/\kappa(J) \le R \le \kappa(J)$ is vacuous here, $\kappa(J) \sim 10^{10}$. (ii) $\mathrm{er}(J)$ is low in absolute terms for both arms, so the separation is relative, not rank-50-vs-1. (iii) $\mathrm{er}(J)$ is the read-map stable rank, a different functional from the write/state participation ratio pr reported in the body (one can move while the other holds); we keep them separate and do not conflate them. This is a geometric, mechanistic result, not a functional recall gap, which the accuracy probes do not show at this scale (Appendix F).

E THE EXTERNAL IMPLEMENTATION, IN DETAIL

This appendix expands the community-implementation result summarised in §5.

A different gate regime. The external model retains moderate momentum, with a median of 0.32–0.52 per chunk. Strong forgetting accompanies near-rank-one writes. The concentration persists when the outer learning rate is halved ($\mathrm{pr} \approx 1.5$). At twice the rate, 8 of 12 blocks become non-finite and the remaining blocks have low participation. Retraining with magnitude coupling keeps write activity nonzero over the sequence and state participation about an order of magnitude higher on all three seeds (Figure 4b). The measured gates change little, which is consistent with an effect through write scaling; it does not identify every change induced by retraining.

Validation and memory activity. The memory-ablated transformer reaches 1.50 nats/char in the original validation pass (byte-level enwik8, so 1 bpc $= \ln 2 \approx 0.69$ nats/char throughout). Across the three learning-rate settings, the retrain scores 1.48/1.61/1.76, versus 2.36/2.36/2.93 for the baseline; these are short-budget, far-from-converged absolute levels, and the retrain differs from the baseline in more than its rank, so the cross-arm gap is not a clean read of collapse. The controlled comparison is within-arm. Each arm changes by less than 0.002 bpc when replayed at the training horizon with writes disabled. These small within-arm differences indicate that new writes contribute little to that particular loss measurement; they do not imply that validation is unable to distinguish the models' overall quality.

We ablate qk-RMSNorm, the adaptive-step cap and spectral normalisation of surprises separately. None becomes non-finite in the first seed tested. Across three seeds for removal of write normalisation, one extinguishes its memory, one becomes non-finite at $8\times$ the training horizon, and one is already non-finite at the training horizon. The extinct seed has zero writes and a zero state within a tenth of the sequence. Its paired short-context loss is close to the control (1.341 versus 1.328 nats/char), rather than better as the unpaired 1.37/1.50 comparison might suggest. The control also has 12.2% fewer parameters (Appendix G). These outcomes show sensitivity to write normalisation, but the mixed extinction/divergence result does not isolate a single causal mechanism.

F SINGLE-QUERY ASSOCIATIVE RECALL

The main prediction experiments do not show a consistent benefit from memory, leaving open whether reduced use is a rational response to the task. A separate associative-recall model, in the style of Arora et al. (2024), tests whether the same family of write rules can help when causal attention sees only its own segment. This is an additional functional test, not a demonstration that collapse is harmful in the original models.

Task. We use the associative-recall generator of `titans.membench`: k key–value pairs (key and value vocabularies of 16) over $2k$ positions, then a QUERY token and a probe key; the model emits the matching value at the last position, the only position scored. With segment size 8, the final segment exposes only about three pairs, so at $k = 16$ thirteen of sixteen queries require the memory and a memory-free model has an approximate reference accuracy of $3/16$ seen plus chance on the rest (≈ 0.24); chance is $1/16 = 0.0625$. At $k = 8$ more pairs fall inside the last segment, so the corresponding local-attention reference is higher.

Model, arms, training. A two-layer `RecallModel` (dimension 64, one head, segment size 8) carrying the gated neural memory studied here. Three arms: the memory-free control (*nomem*, the step forced to 0 so the memory never writes) and two memory arms, the uncoupled Newton–Schulz rule (*unit*) and the same rule with magnitude coupling (*usq*). Each arm trains for 10,000 steps (Adam, batch 64, outer rate 10^{-3}), three seeds, scored by last-position accuracy.

Result. Both memory arms beat the memory-free control at both lengths (median over three seeds):

k	control (nomem)	unit	usq	chance
8	0.460	0.576	0.569	0.0625
16	0.262	0.356	0.345	0.0625

The control scores 0.262 at $k = 16$, close to the approximate local-attention reference of 0.24. Both writing arms improve on it by 0.08–0.12 across the tested lengths. The difference between *unit* and *usq* is smaller than their seed variation, so this experiment does not establish an advantage for either rule. One randomly selected final query can test retention of multiple associations; scoring one answer is not, by itself, a reason to dismiss the result.

Rank and capacity. To connect rank to capacity directly, we retrain the recall model with the memory's usable rank capped to $r \in \{1, 2, 4\}$ (a hard architectural limit, standing in for the near rank-1 state that collapse drives the memory toward) at a fixed outer rate (5×10^{-4}) and up to $50{,}000$ steps. At $k = 16$, the regime where the task is learnable but capacity binds, recall climbs with the cap. Rank 1 and rank 2 stay near 0.19 (three seeds at rank 1, and the rank-1 run carried to $50{,}000$ steps decays from its best back toward chance), while rank 4 reaches 0.47 (chance 0.0625). Restricting the memory toward rank 1 starves its associative capacity. The probe has a ceiling of its own. At $k = 4$ and $k = 8$ every cap saturates, capacity not being the binding constraint, and at $k = 32$ no arm clears chance. Beyond this load the small model does not learn the task, and reading capacity there would need a larger model and a longer schedule.

G Experimental details

Community implementation. The open-source implementation of Wang (2025) is evaluated on byte-level language modelling with enwik8. Architecture, identical across memory arms: model dimension 384, depth 8, segment length 32, sliding-window attention; 4 persistent and 4 long-term memory tokens; neural memories at layers $(2, 4, 6)$, each a depth-4 MLP of width 64, written from 4-token chunks with batched application every 128 tokens; qk-RMSNorm on the write projections; first-order momentum; per-parameter adaptive step with maximum inner rate 0.1; spectral normalisation of surprises on (off in the ablation arm); weight residual between consecutive memories. The propositions of §4 and Appendix C are stated for the two-layer memory of §3; here each memory is a depth-4 MLP with additional implementation details. The generic bounded-write lemma applies wherever its recurrence and assumptions hold, but the two-layer curvature constants and scalar-step stability region are not certificates for this architecture. Memory arms have 20,291,414 parameters; the memory-free control removes the memory layers and the long-term tokens (17,825,640). The control is smaller, and the comparison is not parameter- or compute-matched. Its better loss remains a useful baseline, but does not by itself identify the effect of memory geometry. Training: 512-token windows, batch 1 with 4-step gradient accumulation, AdoptAtan2 (ADOPT (Taniguchi et al., 2024) with the atan2-stabilised update), outer rate 2×10^{-4} (1×10^{-4} / 4×10^{-4} in the sweep arms), 4,000 steps, checkpoints every 1,000. The ψ_c retrain multiplies each Newton–Schulz output by ψ_c evaluated at the write's norm *before* orthogonalisation, with c auto-calibrated per weight shape, an exponential moving average (0.9) of the median pre-orthogonalisation norm, frozen after 50 update calls, and leaves the implementation's spectral normalisation and its 0.1 cap on the inner rate in place. The frozen-η arms replace the momentum head's output by a constant for the whole of training (weights zeroed, bias set to $\operatorname{logit} \eta$, both frozen; we verify post-hoc that the optimiser never moved them). Seed splits worth stating: the write-normalisation removal extinguishes on one seed and is non-finite on two; the frozen-η arms are a single seed each; the three memory-free seeds agree to ± 0.01 bpc at every length.

arm	change vs. baseline	seeds	used in
baseline	community implementation as configured	3	Fig. 4
ψ_c retrain	magnitude coupling during training	3	Fig. 4
LR sweep	outer rate halved / doubled	2×2	Appendix E
no write-norm	surprise normalisation removed	3	Appendix E
memory-free	memory layers and tokens removed	3	Appendix E
frozen η	momentum head pinned (0.5 / 0.9)	1 each	§5

Evaluation: bits per character on fixed validation windows drawn with a fixed seed, so that at a given length every arm sees identical windows; per-position profiles average the loss within eight equal slices of the window. Gates are measured by hooking the three gate heads during a real forward pass on validation text and reporting per-block medians over all chunk-gate values.

Testbed. A compact video world model on EgoLife (Yang et al., 2025a) egocentric video (256-frame clips) compressed to a 128-dimensional PCA latent stream, an episodic item being a 64-latent block: depth-2 model, model dimension 128, segment length 8, prediction horizon 24. Raw-witness runs: the Titans-style recurrence of (1) with $U_t = \theta_t S_t$ and orthogonalisation off, 1,500 steps, batch 8, step-scale ladder, $\alpha_{\text{init}} = 0.002$, $\eta_{\text{init}} = 0.995$. The frozen arms of §5 train for 10,000 steps (30,000-step check), five seeds; chimera and dose interventions run at replay time on frozen twins. The retention probe is that of Figure 1, with distractor gaps counted in blocks and a memory-off rescoring as a within-model ablation. Full configuration files for every arm, the exact commands, and the prediction ledger are described in the supplementary material.

H Additional related work

The survey of §2 covers the 309 works citing Titans at the time of writing, of which 267 full texts were read; we did not identify a general guarantee for the full deep, gated iteration studied here. This survey is not an exhaustive proof of absence, and our own guarantees cover the narrower settings stated in the propositions and lemmas. Wang et al. (2025) unify the fast-weight family as test-time regression over an associative memory, the reading under which our episodic probes are the natural evaluation. Rank collapse is documented for other objects trained by gradient descent. Token representations in pure attention stacks lose rank doubly exponentially with depth (Dong et al., 2021), and contrastive self-supervised embeddings collapse dimensionally (Jing et al., 2022). Both concern the representations of a fixed network; here the collapsing object is a fast weight matrix written at test time, and the driver is what the outer loop learns to feed the write rule.

A related phenomenon is loss of plasticity in networks trained online, where effective rank can fall and units become inactive (Kumar et al., 2021; Lyle et al., 2023; Dohare et al., 2024). Our measurements concern fast weights whose write dynamics depend on outer-trained gates. This suggests a connection, but does not establish the same mechanism or a corresponding loss of functional capacity. Differentiable external-memory systems also motivate the alternative explanation that a model learns to bypass memory when the task does not require it (Graves et al., 2014; 2016).

I Additional figures

Figure 8 gives the mechanism view of the divergence loop of §4, on the runs of Figure 2.

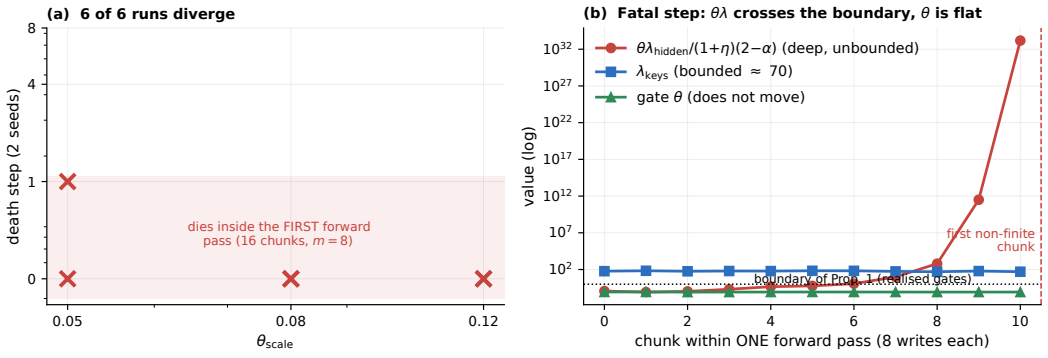

Figure 8: **The feedback loop, mechanism view** (same runs as Figure 2). *(a)* Death step per step-scale rung (two seeds each; boundary rung $\theta_{\text{scale}} = 0.03 \approx \theta^*$ excluded). All six supra-threshold runs diverge, and for $\theta_{\text{scale}} \geq 0.05$ most runs die inside the *first* forward pass of 16 chunks ($m = 8$). *(b)* The fatal step of one run ($\theta_{\text{scale}} = 0.12$), chunk by chunk. The key-side curvature stays near 70 (its bound in (2) is $2\|k\|^2\|W_2 \operatorname{diag} \sigma'\|_2^2$, so key normalisation pins $\|k\|$ while $\|W_2\|$ stays finite here and W_1 is what diverges), and the learned gate θ is flat, while $\theta\, \lambda_{\text{hidden}}$ crosses its realised-gate diagnostic threshold before the state becomes non-finite a few chunks later. Key normalisation alone does not bound this hidden-side Gauss–Newton term as the fast weights change.