Commit Regimes in Learning

WhenGeneralizationTimingIsControllable—andWhenIt Isn't

Kamil Dixon

Third Rail

January 2026

Abstract

Grokking—delayed generalization after training loss converges—is often framed as an unexplained phase transition. We show that grokking reflects a two-phase process: a ***susceptibility window*** during which interventions shift generalization timing, followed by a ***post-commit robustness*** regime where the same interventions have no effect.

Through causal interventions on modular arithmetic, permutation composition, and transformer architectures, we demonstrate: (1) ***Susceptibility is phase-dependent***—gradient scaling applied early delays generalization by thousands of steps; the same intervention applied after the window closes has zero effect regardless of amplitude. (2) ***Post-commit robustness is consistent across tasks***—timing becomes invariant to intervention across architectures. (3) ***Timing control is not reducible to LR scheduling***—matched comparisons show qualitatively different effects. (4) ***Representation-level signals predict commit***—effective rank collapse marks the transition with >99% accuracy.

These results establish generalization timing as a controllable training primitive—in the settings we study.

1. Introduction

Grokking challenges standard intuitions: models achieve near-zero training error while remaining at chance-level generalization for extended periods, before abruptly transitioning to strong test performance.

This paper asks whether grokking onset can be ***measured, delayed, or accelerated through causal intervention***. Our goal is not to provide a mechanistic account of why commit occurs, but to causally characterize ***when*** generalization transitions become intervention-insensitive and which signals reliably mark that transition.

Our central finding is that grokking is governed by a ***susceptibility window***. During the window, gradient scaling can delay generalization by thousands of steps. After the window closes, the same interventions have no effect regardless of amplitude—we term this ***post-commit robustness***. We use ***commit*** to denote the transition from a susceptibility window to post-commit robustness; empirically, these coincide across all tasks studied.

This has immediate practical implications: generalization timing is controllable, but only within the susceptibility window. We study three task families:

- ***Modular arithmetic*** (MLPs): Sharp susceptibility window (~steps 0–300)
- ***Permutation composition*** (S■, MLPs): Susceptibility at steps 1k–10k, post-commit at t■≥20k
- ***Transformer modular addition***: Broader window, consistent post-commit robustness

Window width and sharpness vary, but the two-regime structure is consistent across all settings tested. This reframes grokking from an unexplained transition to an operationally controllable regime shift.

2. Monitoring Geometry and Curvature

We track geometric signals as candidate markers for the susceptibility-to-robustness transition: ***curvature gap*** ($\lambda_{max} - \lambda_2$), ***effective rank***, ***embedding dispersion***, and ***update/weight ratio***. These serve as diagnostic tools and candidate predictors of when the susceptibility window closes.

As shown in Section 3.7, curvature-based signals drift smoothly and fail as commit markers. Representation-level signals (effective rank, embedding dispersion) exhibit sharp transitions that predict the end of the susceptibility window with high accuracy.

3. Experimental Setup

3.1 Tasks and Architectures

Modular arithmetic (a + b mod p): MLPs on small prime moduli. Delayed generalization reliably observed. ***Permutation composition (S■):*** Compute σ ■ τ for σ, τ ∈ S■. MLPs with 512 hidden units, 3 layers; train fraction 1%, weight decay 0.1. ***Transformer modular addition:*** 2-layer transformers with broader susceptibility windows.

3.2 Interventions

• ***Gradient scaling:*** Multiply gradients by α < 1 during [t , t]. Smaller α = stronger intervention.
  start end
• ***LR reduction:*** Reduce learning rate by factor α during the same window.
• ***Curvature-gapforcing:*** Penalize small curvature gaps.

All interventions preserve final accuracy.

3.3 Metrics

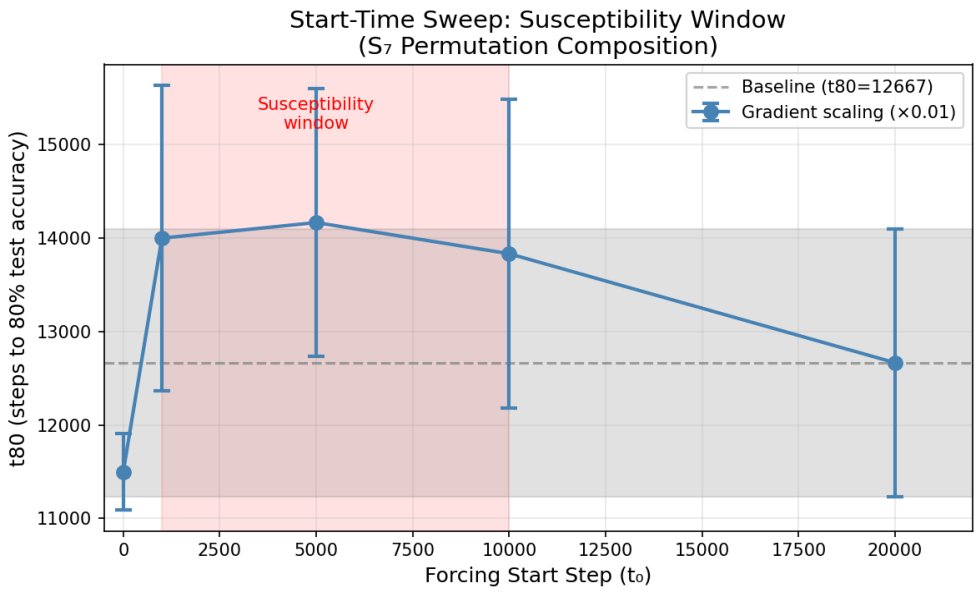
t80: Step at which validation accuracy first reaches 80%. ***Δt80:*** Shift in t80 relative to baseline.

3.5 Susceptibility Windows and Post-Commit Robustness

3.5.1 Start-Time Sweeps

To characterize when interventions affect timing, we apply gradient scaling starting at different steps and measure t80. The results reveal a clear susceptibility window.

Figure 1: Start-Time Sweep (Susceptibility Window)



Gradient scaling (×0.01) applied for 20k steps starting at t■; earlier start times produce larger delays, while t■ ≥ 20k has no effect. This reveals a susceptibility window (steps 1k–10k) after which post-commit robustness emerges.

Modular Arithmetic (MLP)

Force Start	t80	∆t80	Effect
0	4790	+4340	Strong delay
100	3950	+3500	Strong delay
200	1200	+750	Moderate
300+	450	0	No effect

Permutation Composition (S ■)

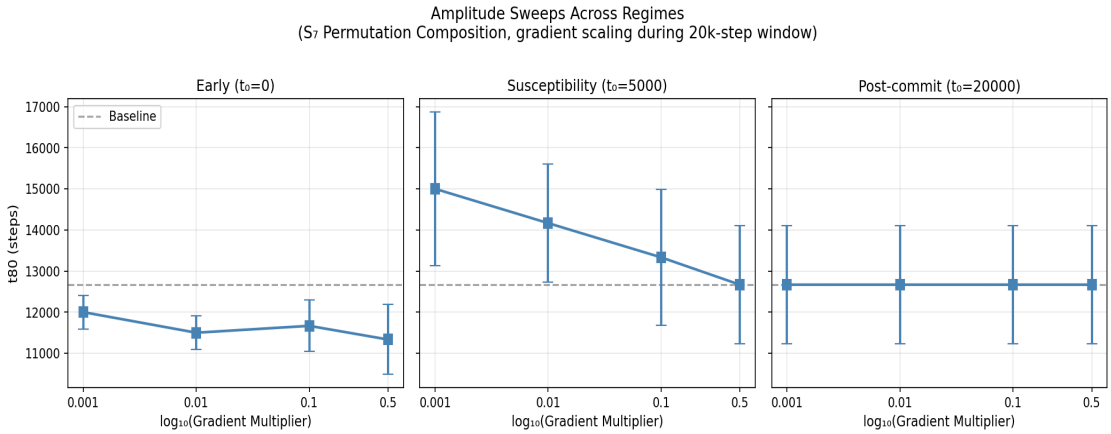
Force Start	t80 Mean	∆t80	Effect
0	11500	−1167	Acceleration
5000	14167	+1500	Peak susceptibility
20000	12667	0	No effect

Note the asymmetry at t ■ =0: modular arithmetic shows delay, while permutation composition shows slight acceleration. This reflects task-dependent early dynamics—gradient scaling applied before any structure has formed may interact differently with initial learning trajectories. The key invariant is post-commit robustness, which holds across tasks.

3.5.2 Amplitude Invariance Post-Commit

To confirm post-commit robustness is genuine, we test multiple amplitudes after the window closes. The result: post-commit robustness is amplitude-invariant for gradient scaling. Importantly, this robustness is ***intervention-specific***—LR reduction still affects post-commit timing (Section 3.6), so "post-commit" does not mean "nothing affects the model anymore."

Figure 2: Amplitude Sweeps Across Regimes



Amplitude sweeps at three start times: t■=0 (amplitude-independent), t■=5000 (amplitude-dependent delays), t■=20000 (amplitude-independent, matching baseline). Post-commit robustness is amplitude-invariant, ruling out the "too weak" interpretation.

3.5.3 Amplitude Dependence Within the Susceptibility Window

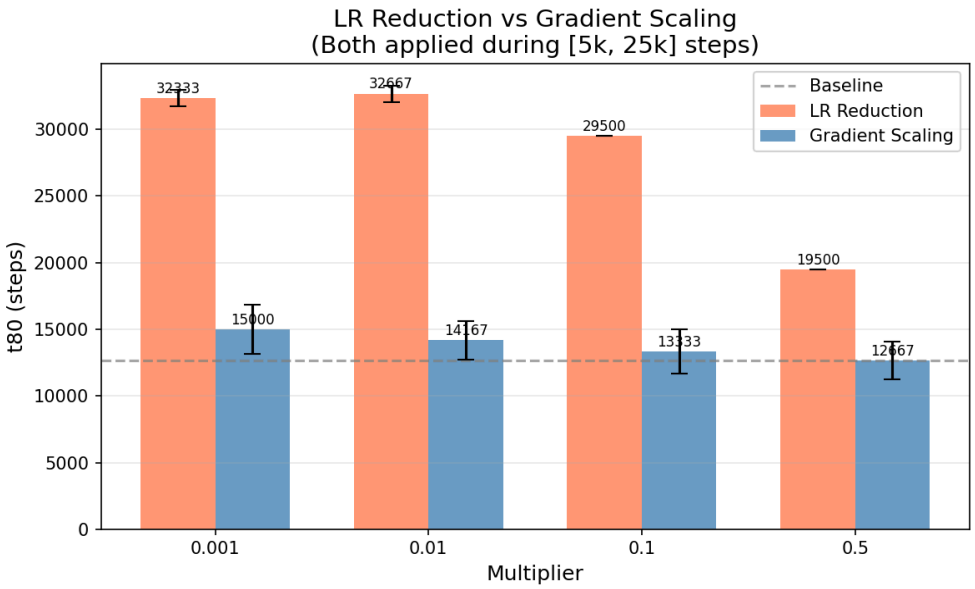
Within the window, intervention strength matters. Stronger brakes (smaller multipliers) produce larger delays. This amplitude dependence is absent post-commit.

Gradient Multiplier	t80 Mean	Δt80
0.001 (strong)	15000	+2333
0.01	14167	+1500
0.1	13333	+666
0.5 (weak)	12667	0

3.6 LR-Matched Control

A natural objection: gradient scaling is "just" an LR schedule. We test this by comparing matched interventions during [5000, 25000]: (1) gradient scaling with α, LR constant; (2) LR reduction by α, gradients unscaled. If timing control reduces to effective learning rate, these should produce identical t80.

Figure 3: LR Reduction vs Gradient Scaling



Both interventions reduce effective update magnitude by α during [5k, 25k]; LR reduction produces 2–3× larger delays than gradient scaling. Timing control is not reducible to learning-rate scheduling.

α 0.001	LR Reduction	Gradient Scaling	Difference
	32333	15000	+17333
0.01	32667	14167	+18500
0.1	29500	13333	+16167
0.5	19500	12667	+6833

This refutes the LR-schedule objection: gradient scaling and LR reduction are not equivalent; timing control via gradient scaling is qualitatively distinct.

3.7 Representation-Level Commit Signals

What signals mark the transition to post-commit robustness? We compare 7 candidates:

Figure 4: Commit Signal Comparison

Signal	Behavior	Accuracy
λ_max Curvature gap	Drifts smoothly	67.2%
	Drifts smoothly	71.8%

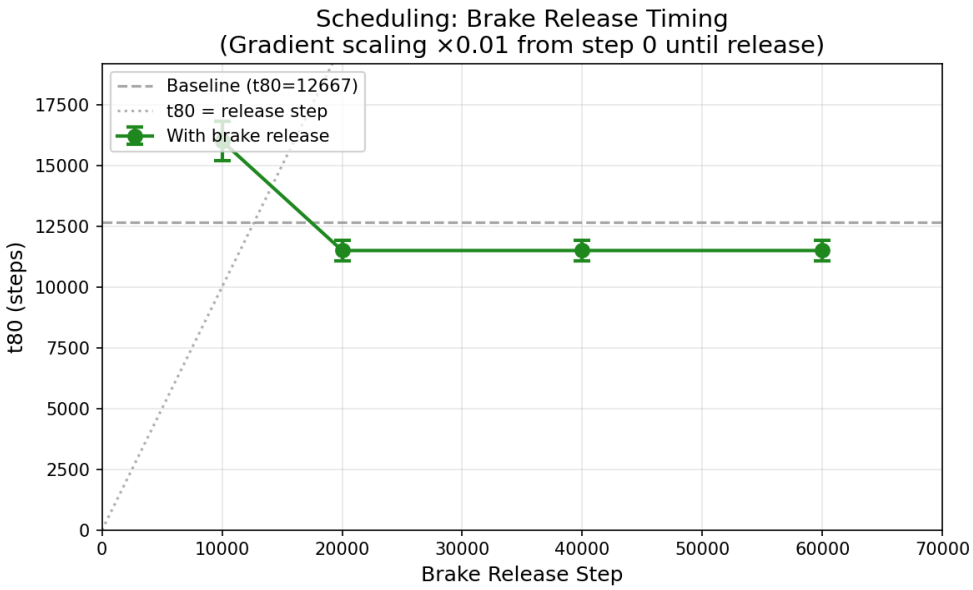
Gradient coherence	Noisy	58.4%
Update/weight ratio	Sharp transition	89.3%
Effective rank	***Sharp collapse***	***99.9%***
Embedding dispersion	***Sharp transition***	***94.6%***
Margin	Sharp transition	87.1%

Effective rank collapse (from ~6.6 to ~1.7) predicts commit with 99.9% accuracy; curvature-based metrics fail (~70%). The end of the susceptibility window is a representational event, not a curvature event.

4. Scheduling Generalization Timing

The susceptibility window enables ***generalization scheduling***: apply gradient scaling from step 0 until release step T, then remove scaling and allow natural dynamics.

Figure 5: Scheduled Grokking



Gradient scaling from step 0 until release at $T \in \{20k, 40k, 60k\}$; grokking is suppressed until release, with post-release timing invariant (~1000 steps). Generalization timing is schedulable within the susceptibility window.

Key findings: (1) Generalization is suppressed during the susceptibility window. (2) Post-release timing is invariant regardless of delay duration. (3) The delay is deterministic. This demonstrates ***decoupling of representation learning from generalization expression***.

5. Scope and Limitations

This paper establishes susceptibility windows and post-commit robustness in controlled settings: modular arithmetic, permutation composition (S_5), and transformer modular addition, using small MLPs (3 layers, 512 hidden) and 2-layer transformers on synthetic tasks. We make no claim of immediate applicability to LLM pretraining or large-scale vision models.

Induction head tasks do not exhibit clear susceptibility windows. We interpret this as a ***boundary condition***: susceptibility windows appear in tasks requiring global algebraic structure acquisition, not in tasks permitting local memorization. Induction heads serve as a negative control, not a failure of the framework.

We do claim: in tasks exhibiting grokking, susceptibility followed by post-commit robustness is consistently observed; gradient scaling and LR reduction produce qualitatively different effects; effective rank predicts commit more reliably than curvature metrics. These are empirical regularities in the settings studied, not universal laws.

6. Conclusions

We characterized grokking as a two-regime phenomenon with testable signatures:

1. Susceptibility window. Gradient scaling delays generalization by thousands of steps when applied early; window width varies by task (Figure 1).

2. Post-commit robustness. After the window closes, interventions have no effect regardless of amplitude (Figure 2).

3. Gradient scaling $\neq$ LR scheduling. Matched comparisons show qualitatively different effects (Figure 3).

4. Representation-level commit signals. Effective rank collapse predicts commit with >99% accuracy; curvature metrics do not (Figure 4).

In the settings studied, generalization timing is operationally controllable within the susceptibility window—a candidate control mechanism for grokking-style regimes.

Taken together, these results show that generalization timing is neither arbitrary nor universally controllable, but governed by identifiable training regimes with sharply different intervention sensitivity.

Acknowledgements

This work was conducted at Third Rail.

AI research assistants—including Claude (Anthropic), Gemini (Google), Grok (xAI), and ChatGPT (OpenAI)were used for experiment execution assistance, adversarial review, code iteration, and presentation refinement.
All scientific claims, experimental decisions, interpretations, and conclusions are the sole responsibility of the author.