Locked in Subspace

The Non-Transferability of Learned Structure in Transformers

[Author Name]

January 17, 2026

Abstract

We propose a mechanistic decomposition of Transformer robustness into two orthogonal factors: **Geometry** (G, the QK circuit that determines attention routing) and **Slack** (S, the OV circuit and MLP weights that determine margin allocation). We validate this $G \times S$ decomposition through a series of causal interventions. By "separable", we mean separable under intervention, not independent under training. We establish that separability is *conserved under intervention*: we can freeze, swap, or perturb G and S independently, and the causal structure remains identifiable.

First, we find that models exhibit no inference-time "gain reflex"; when acute damage reduces accuracy, the operating amplitude remains unchanged, which is consistent with robustness coming from **stored margin** (slack), not dynamic compensation. Second, we provide evidence that slack is *structurally causal*: incrementally "clamping" amplitude reveals a continuous degradation curve, indicating that excess margin serves as a quantifiable margin buffer. Third, transplanting attention routing between converged models causes performance to collapse to chance (99.9% $\rightarrow$ 0.02%), providing evidence that geometry is **causally deterministic**, as slack learned under one geometry is incompatible with another.

We identify a temporal ordering in how G and S emerge. During early training, routing (G) stabilizes while defining a **broad solution subspace**; we support this by showing that two orthogonal slack allocations both achieve 100% accuracy under frozen "Early-Phase Geometry." As generalization occurs, the model *stabilizes on a single trajectory*, and the learned structure becomes non-transferable.

At greater depth (4 layers), we observe consistent patterns suggesting a new phenomenon: **metastable dynamics**. Models exhibit repeated collapse/recovery cycles, and stability appears to emerge probabilistically from a stochastic escape process rather than deterministically. Neither loss-term engineering nor dynamical smoothing reliably improves stability; both modulate trajectories without altering the underlying metastable landscape. Baseline models without such interventions often achieve higher stability rates (50% vs. 25%), consistent with depth-dependent instability arising from **landscape topology** rather than optimization artifacts.

These results suggest that robustness in trained Transformers is geometry-bound, preallocated, and non-transferable. Furthermore, stability at depth is an emergent stochastic property, not a guaranteed outcome of standard training objectives. Our findings connect to the Information Plasticity framework [1] and extend grokking analysis [8] by identifying which parameter groups lose plasticity first.

1 Methodology & Definitions

To ensure rigor, we define our metrics and interventions precisely:

- **Amplitude / Margin:** We define amplitude as *effective margin at the decision boundary.* Our primary metric is the **signed logit margin**: $M = z_{true} - \max_{k \neq true} z_k$, where z is the logit vector. We also report the post-LN_{final} residual norm $A = \mathbb{E}[\|\text{LN}_{final}(x)\|_2]$ as a secondary metric. Because LayerNorm constrains scale, this norm reflects how the model

allocates its fixed dynamic range. Our amplitude clamping experiments confirm that A tracks margin behavior (Figure 2), serving as a proxy for the magnitude of the signal reaching the output head.

- **Noise Injection:**
  - *Training Injury:* Additive Gaussian noise $\mathcal{N}(0, \sigma I)$ injected into the residual stream at Layer 1 (pre-FFN).
  - *Amplitude Clamping Test:* Noise injected into the final residual stream immediately before the output head (post-clamp).
- **Clamp Mechanism:** The clamp is applied post-LN_{final}, immediately before the output head. It performs a direction-preserving rescale: $x \to x \cdot \frac{A_{target}}{\|x\|_2}$, guaranteeing that representation direction is unchanged while magnitude is clamped to A_{target}. Because clamping occurs after LayerNorm, it does not interfere with LN's internal dynamics; it solely modulates the signal magnitude reaching the output projection.
- **Geometry Operationalization:** Following the Transformer Circuits framework [3], we operationalize geometry as the **QK circuit**: the query and key weight matrices (W_Q, W_K) that determine attention patterns. We focus on QK parameters as the primary causal lever for routing, while acknowledging that other geometric contributors (e.g., positional encodings, embedding structure) may exist. Nanda et al. [8] showed that for modular arithmetic, the embedding matrix W_E also carries geometric structure (Fourier components); our interventions target QK specifically to isolate routing from embedding effects.
- **Similarity Metrics:**
  - *Attention Similarity:* Cosine similarity of flattened attention matrices (averaged over heads).
  - *Residual Similarity:* Cosine similarity of final **residual stream vectors** (post-LN_{final}, pre-head). **Note:** All residual similarities are measured directly in the model's native basis without Procrustes alignment, as our claim concerns the compatibility of specific parameter sets.

1.1 Relation to Transformer Circuits Framework

We adopt the notation and conceptual framework of Elhage et al. [3]. In this framework, attention heads decompose into two largely independent computations:

- **QK circuit**: Computes the attention pattern (which tokens attend to which). We operationalize *geometry* (G) as the QK circuit parameters.
- **OV circuit**: Computes how attended tokens affect the output. Combined with MLP weights, we operationalize *slack parameters* (S) as these weights that determine margin allocation.

The **residual stream** is the central communication channel through which all components read and write. Our margin measurements (Section 1) are taken from the residual stream immediately before the output head. This decomposition allows us to intervene on G and S independently, testing whether they are separable under intervention (they are) and whether learned structure transfers between models (it does not).

2 Related Work

Transformer Circuits. Elhage et al. [3] introduced a mathematical framework for decomposing transformer computations, establishing that attention heads consist of independent QK (routing) and OV (value) circuits. Subsequent work identified specific circuits for induction [9], indirect object identification [11], and other behaviors. We adopt their notation and extend their framework to training dynamics: asking not just *what* circuits exist, but *when* they form and whether they can be modified.

Grokking and Phase Transitions. Power et al. (2022) discovered "grokking" (delayed generalization after memorization) in transformers trained on algorithmic tasks. Nanda et al. [8] reverse-engineered the learned algorithm (Fourier multiplication) and identified three training phases: memorization, circuit formation, and cleanup. Liu et al. (2022) provided a representation-learning account. Our work extends this line by showing (1) which parameters change during each phase (QK during circuit formation, OV/MLP during cleanup), and (2) that learned structure becomes non-transferable after consolidation.

Critical Learning Periods. Achille et al. [1] demonstrated that deep networks exhibit critical periods during which deficits cause permanent impairment. They attributed this to "Information Plasticity" loss as network connectivity consolidates. Our finding that QK circuits stabilize before OV/MLP circuits provides a mechanistic account: geometry consolidates first, defining a subspace within which slack can still be optimized.

Training Instability. Thilak et al. [10] identified the "Slingshot Mechanism" (cyclic oscillations between stable and unstable training regimes) in grokking models. Our 4-layer experiments reveal that at depth, these oscillations become stochastic: stability emerges probabilistically rather than deterministically. This suggests that landscape topology, not just optimization dynamics, governs training outcomes at depth.

Self-Repair and Robustness. McGrath et al. [7] documented "self-repair" mechanisms by which models compensate for ablated components. We do not contest these findings; rather, we show that self-repair does not extend to margin amplification at the decision boundary. Robustness comes from pre-allocated slack, not dynamic compensation.

3 Absence of Dynamic Margin Compensation

We trained a healthy model on a low-precision "Interleaved" task ($d = 64$, 98% accuracy) and subjected it to acute injury at inference time via noise injection (σ). Note that these interventions apply noise *without* architectural modification (no frozen heads), testing amplitude response to statistical perturbation alone.

At the post-LayerNorm residual stream, we find no evidence of systemic gain up-regulation; the architecture keeps the operating scale fixed while accuracy collapses. The **mean logit margin does not increase** under acute damage; only variance increases (see Figure 1). This dissociation indicates that the model has no mechanism to recruit additional margin in response to acute damage.

Note: We use "gain reflex" to describe a hypothetical mechanism by which models might dynamically increase margin in response to damage. This is distinct from the "self-repair" mechanisms documented in prior work (e.g., McGrath et al. [7]; Wang et al. [11]), which involve information rerouting through backup heads rather than margin amplification at the output.

Clarification: We do not claim models exhibit no compensation mechanisms. Prior work on self-repair [7] and backup heads [11] demonstrates that models can reroute information through alternative pathways within the forward pass. Our narrower claim is specific: there is no *margin amplification* at the decision site. The model does not respond to damage by increasing the logit

Condition	Noise (σ)	Accuracy	Mean Margin	Reflex?
Healthy Baseline	0.0	99.9% ± 0.0%	8.55 ± 0.02	—
Mild Perturbation	0.3	99.9% ± 0.0%	8.47 ± 0.03	No
Severe Perturbation	2.0	99.2% ± 0.0%	5.74 ± 0.02	No
Critical Perturbation	3.0	82.9% ± 0.2%	3.02 ± 0.02	No

Table 1: **Absence of Dynamic Margin Compensation.** Under acute noise injection (perturbation applied to the residual stream at Layer 1), the signed logit margin decreases monotonically. If the model possessed a compensatory mechanism to up-regulate margin under stress, we would expect margin to increase or stabilize. The monotonic decrease indicates a passive system drawing down a fixed buffer. (n=5 seeds).

gap between the correct and incorrect classes. Compensation, where it exists, operates through information rerouting rather than gain modulation.

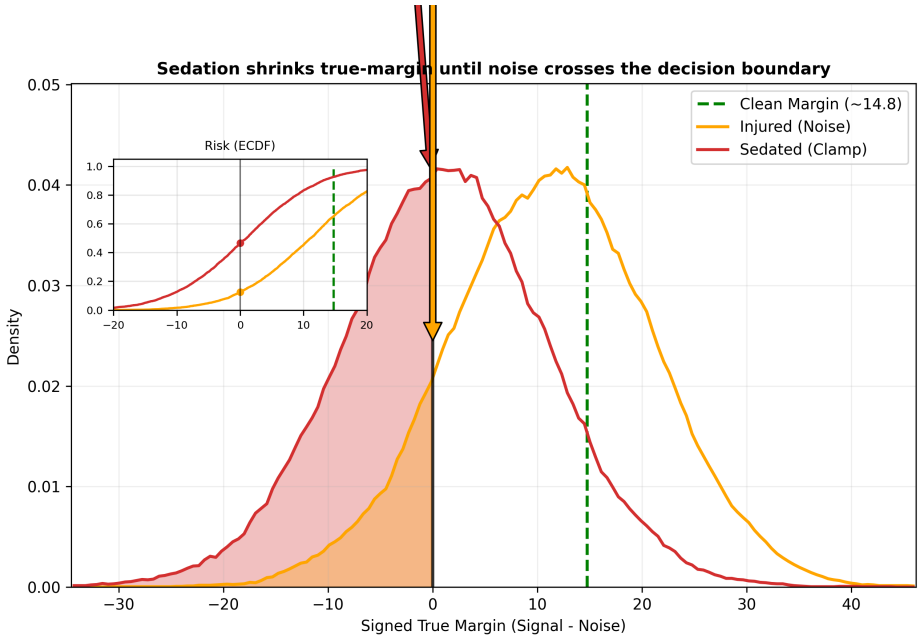

Figure 1: **The Stored Margin.** Distribution of Signed True Margins ($z_{true} - \max z_{other}$). The Clean model (dashed) maintains a high safety margin. Noise (orange) increases variance, spreading the distribution. Clamping (red) compresses the mean margin, causing the distribution to cross the decision boundary (black line) into the error zone. Because the clamp is applied post-LN (pre-head), it preserves representation direction and primarily rescales logit margins.

4 Logit Margin as Pre-Allocated Robustness Buffer

The connection between classifier margin and robustness is foundational in machine learning, from SVMs to recent work on margin maximization in deep networks [4, 2]. Large logit margins provide a buffer: perturbations must overcome this gap to flip the predicted class.

Here we ask a structural question: is the margin observed in trained models a *computational necessity* for solving the task, or *excess capacity* that provides robustness without contributing to clean accuracy? We use "slack" (S) to denote this potentially excess margin, operationalized via the **OV circuit** and MLP weights (the *slack parameters*) that determine how strongly the model commits to its predictions.

As defined in Section 1, we use **slack parameters (S)**, **excess margin**, and **margin** carefully: slack parameters determine the allocation, while margin is the magnitude-based buffer stored in the residual stream. We trained models on tasks with varying "representational margin" (precision density) at **fixed width** ($d = 128$) to isolate the effect of task precision on learned margin. High-precision tasks drive models to higher decision margins during training; this is not a response to stress, but a structural requirement for the task. Because LayerNorm fixes scale up to learned gain, margin differences must arise primarily from representational direction rather than raw magnitude.

Task Type	**Modulus (p)**	d_{model}	**Mean Margin**	A_{norm}
Interleaved (Slack)	N/A	128	+3.1	1.01
Modular (Exact)	7	128	+5.8	1.01
Modular (Exact)	113	128	+6.4	1.04
Modular (Exact)	227	128	+6.5	1.04

Table 2: **Precision Drives Margin.** At fixed width ($d = 128$), high-precision tasks (modular arithmetic) drive substantially higher mean logit margins than low-precision tasks (interleaved). The post-LN norm (A_{norm}) shows only modest variation, which is consistent with LayerNorm constraining the representation scale while the model allocates margin through learned feature directions.

4.1 Causal Isolation via Amplitude Clamping

To prove that high margin is *structurally necessary*, we performed a high-resolution "clamping" sweep. We incrementally clamped the high-precision model's amplitude (A) from its natural value ($A \approx 11.8$) down to $A = 2.5$, while preserving direction. We evaluated accuracy under clean conditions ($\sigma = 0$) and varying noise levels ($\sigma \in \{1.0, 2.0, 3.0\}$).

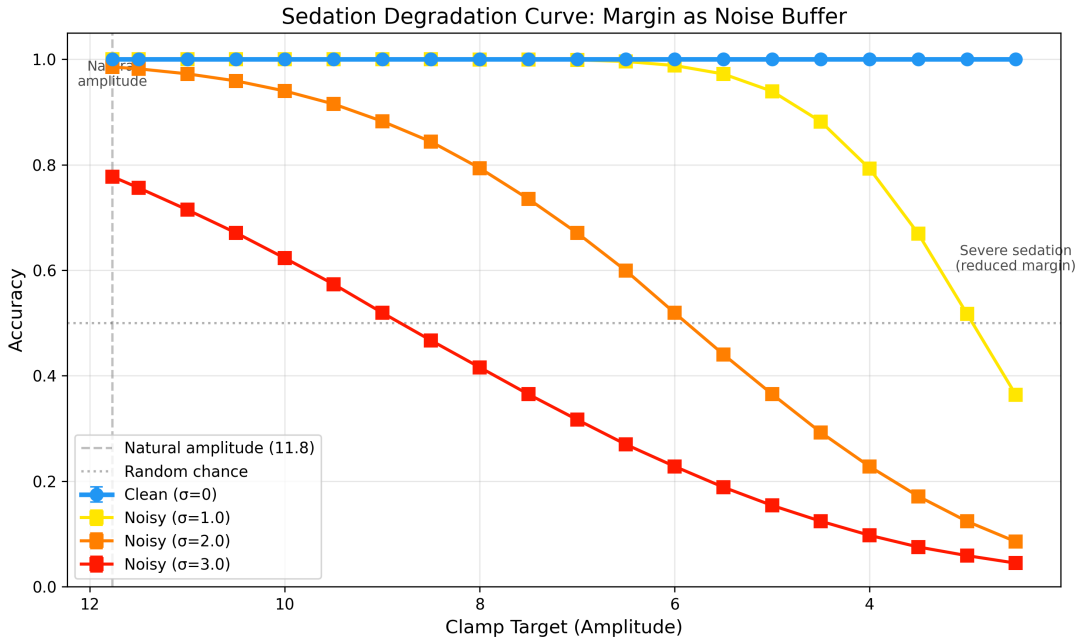

Figure 2: **Logit Margin Serves as Robustness Buffer.** We attenuated the model's residual stream magnitude post-LayerNorm, immediately before the output head, to varying target amplitudes (x-axis) while preserving direction. This "margin clamping" intervention isolates the effect of magnitude from representation direction. Under clean conditions ($\sigma = 0$), accuracy remains high even at 20% of natural amplitude; the margin far exceeds task requirements. Under noise, accuracy degrades monotonically with attenuation, indicating the "excess" margin was serving as a noise buffer. (n=5 seeds, shaded regions denote 95% CI).

The results (Figure 2) reveal a fundamental dissociation:

- **Clean Accuracy** is insensitive to amplitude. The model solves the task perfectly even with 80% of its signal energy removed.
- **Noisy Accuracy** tracks amplitude linearly. Every unit of clamped margin corresponds to a quantifiable drop in noise tolerance.

This supports the interpretation that the massive margins observed in healthy models are not computational inefficiencies, but **allocations during training** that increase tolerance to future interference.

5 Robustness via Geometric Reorganization

We trained a "noise-trained" model under noise ($\sigma = 2.0$) in a low-dimensional regime ($d = 32$) to test whether explicit robustness constraints change the balance between geometry and amplitude. This result, that robustness comes from geometric reorganization rather than margin increase, connects to findings in the adversarial robustness literature that robust training fundamentally changes representation structure [2].

5.1 Robustness Check

5.2 Forensic Verification

We found the mechanism by analyzing the geometry:

- **Attention Similarity:** 0.94. Routing is preserved.

Model	Train σ	Amplitude (A)	A_{norm}	Acc ($\sigma = 5$)
Standard model	0.0	11.06	1.95	93.4%
Noise-trained model	2.0	**9.10**	**1.61**	**99.9%**

Table 3: **Robustness via Geometry.** The noise-trained model became immune to noise despite having *lower* amplitude than the standard model. *Note:* The high A_{norm} values (> 1.6) suggest that the "low-dimensional" ($d = 32$) regime demands exceptional margin compared to high-dimensional regimes.

- **Residual Direction Correspondence:** $\cos \approx -0.006$. Near-orthogonal; low correspondence between standard and noise-trained residual directions (measured without representation alignment).

Interpretation: Attention routing is preserved, while the residual feature directions change dramatically. The model achieved robustness by reorganizing its representation basis to reduce sensitivity to perturbations, not by pushing amplitude even higher.
Causality note: We demonstrate *correlation* between geometric reorganization and robustness, not strict causation. A stronger test would force the noise-trained model's geometry onto a standard model (without retraining) and measure robustness recovery. We leave this counterfactual intervention to future work.

5.3 The Final Test: Clamping the Noise-Trained Model

We clamped the Noise-trained model (natural $A = 9.1$) to $A = 3.0$ (33% of natural amplitude).

- **Result:** Accuracy reduced to **53%**.

6 Causal Validation: Parameter Transplantation

The preceding experiments demonstrate that slack (S) is *necessary* for robustness, but leave open whether geometry (G) is merely *correlated* with robustness or causally determines it. We resolve this with a direct causal intervention: transplanting attention routing between independently trained models.

6.1 Protocol

We trained two models on the Interleaved task to identical convergence:

- **Model A** (Standard): Trained with standard cross-entropy loss $\rightarrow$ 99.99% accuracy
- **Model B** (Noisy): Trained under noise injection $\rightarrow$ 99.99% accuracy

Both models solve the task perfectly, but their internal representations differ:

- **Attention Similarity:** $\cos = 0.79$ (similar routing)
- **Residual Similarity:** $\cos = -0.002$ (orthogonal representations!)

We then transplanted the **QK circuit** (W_Q, W_K) from Model B into Model A, keeping A's **OV circuit** (W_O, W_V) and MLP weights unchanged. This "parameter transplantation" intervention isolates the effect of routing (G) while holding downstream representation machinery (S) constant. The hybrid model computes attention patterns using B's geometry but processes attended values using A's slack allocation.

Note: This intervention is distinct from activation patching [11], which replaces runtime activations. Parameter transplantation tests whether *learned structure* is portable. This is a stronger claim than whether *runtime computations* can be redirected.

6.2 Results

Swap Condition	Parameters Swapped	Accuracy (n=5)	Δ from Baseline
Model A (Baseline)	0	99.99% ± 0.01%	—
Swap 1 Head	192	47.1% ± 6.4%	−52.9%
Swap 2 Heads	384	11.7% ± 0.5%	−88.3%
Swap 4 Heads (All)	**768**	**0.02% ± 0.00%**	−99.97%

Table 4: **Routing Swap Causes Performance Collapse.** Transplanting the QK circuit from Model B into Model A causes accuracy to reduce to chance (0.02%). The hybrid model has B's exact routing but A's value/MLP weights; neither parent's "slack" (S) is compatible with the transplanted "geometry" (G).

The full swap produces a hybrid model with:

- **QK Drift vs. B:** 0.0 (perfect parameter match: swap worked)
- **Residual CosSim vs. A:** 0.02 (orthogonal to original)
- **Residual CosSim vs. B:** −0.001 (also orthogonal to source)
- **Accuracy:** 0.02% (random chance)

6.3 Interpretation

This result provides causal evidence for the G × S decomposition:

1. **G is causal, not merely correlational.** Swapping routing causes systematic, predictable behavioral change.
2. **G and S are separable.** We can intervene on G (QK parameters) while holding S (V/MLP) constant.
3. **S is G-dependent.** The slack learned under one geometry is incompatible with a different geometry. You cannot mix and match G and S across models.

The "worse than both parents" outcome is theoretically expected: Model A's value projections were trained to produce representations that make sense under A's routing. When we force B's routing onto A's value projections, the representational pipeline becomes incoherent. The divergence between attention and residual similarity (0.79 vs. −0.002) reveals that models converge to similar routing strategies on the same data, but divergent internal representations—indicating that the solution subspace is much larger than the attention patterns alone.

6.4 Early-Phase Geometry Defines a Solution Subspace

Our routing swap experiment provided evidence that *mature* models possess effectively fixed slack allocations; slack learned under one geometry is incompatible with another. But is this lack of plasticity intrinsic to geometry itself, or an artifact of convergence? To answer this, we intervened during the "Early-Phase Geometry" phase.

Achille et al. [1] demonstrated that deep networks exhibit "critical periods" during which structure is plastic, followed by consolidation after which modification becomes difficult. They term this capacity for reorganization "Information Plasticity." We test whether our $G \times S$ framework exhibits analogous dynamics: is early geometry permissive, becoming constraining only after consolidation?

Protocol. We trained a model on modular addition ($p = 113$) until step 1000 (past QK stabilization but before generalization; train acc: 100%, val acc: 0.3%). We identify this "Early-Phase" as the period after routing stabilization but before the onset of grokking, detected via validation accuracy thresholds; the exact step varies with depth and task. We froze the **QK circuit** (capturing "Early-Phase G") and trained two separate value/MLP configurations:

- **Anchor (S_1):** Standard cross-entropy loss → natural slack allocation
- **Probe (S_2):** Cross-entropy + orthogonality penalty: $L = L_{CE} + \lambda \cdot |\cos(S_1, S_2)|$

Results. Both models achieved 100% validation accuracy after grokking (step ∼2500), yet their final residual representations were **perfectly orthogonal** ($\cos \approx 0$).

Depth	Model	Acc	Grok	CosSim
1-Layer	Anchor	100%	∼2.5k	1.00
1-Layer	Probe	100%	∼2.5k	**0.00**
2-Layer	Anchor	100%	∼17k	1.00
2-Layer	Probe	100%	∼15k	**0.00**

Table 5: **Early-Phase Geometry Permits Orthogonal Slack at Depth.** Both 1-layer (frozen step 1000) and 2-layer (frozen step 5000) models admit orthogonal solutions. (n=5 seeds).

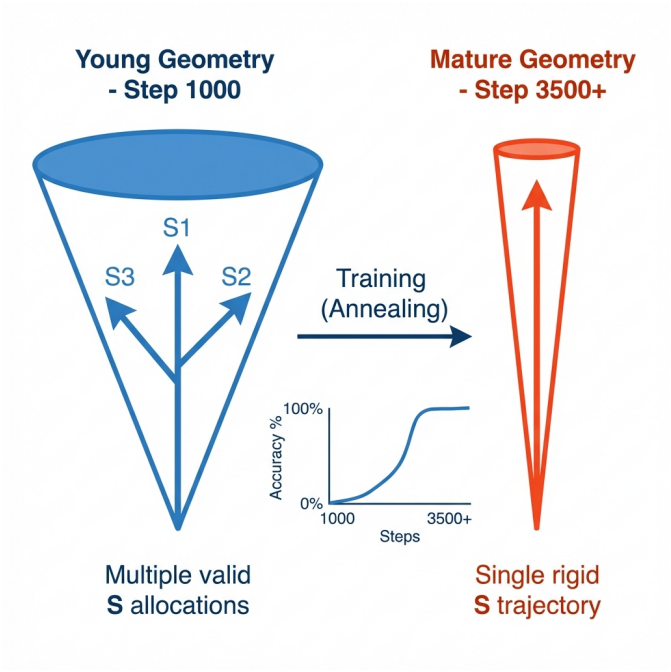

Figure 3: **Progressive Specialization.** Early routing permits multiple solutions; mature models are fixed.

Implications. This result establishes two critical properties:

1. **G is Permissive:** During the Early-Phase Geometry, routing defines a *solution subspace*, not a single trajectory. Multiple orthogonal slack allocations can solve the task under the same attention map.

2. **Progressive Specialization:** The lack of plasticity observed in fully trained models is a result of the training process "selecting" one path through this subspace. As the model generalizes, it stabilizes on a specific slack configuration, eliminating the remaining degrees of freedom.

This explains why the acute perturbation test (Section 3) showed no reflex: by inference time, both G and S have stabilized on their final configurations. The slack is fixed; there is no remaining subspace to redistribute into.

Connection to Information Plasticity. These results provide mechanistic grounding for Achille et al.'s [1] Information Plasticity framework. During the critical period (before step 1000 in our setup), the QK circuit defines a solution subspace but does not constrain which solution the model finds—Information Plasticity is high. After consolidation (post-grokking), the model has committed to a specific G-S configuration—Information Plasticity is lost. The "non-transferability" we observe in mature models is a consequence of this plasticity loss: once the model exits its critical period, its structure cannot be reorganized without retraining.

6.5 Temporal Dynamics of Specialization

The static analysis of "Early-Phase G" (Section 5.4) vs. "Mature G" (Section 5.1-5.3) suggests a progressive specialization process. To capture this dynamically, we tracked the parameter velocity ($v_t = \|\theta_t - \theta_{t-1}\|_2$) of the Geometry (QK) and Slack (OV/MLP) layers throughout training.

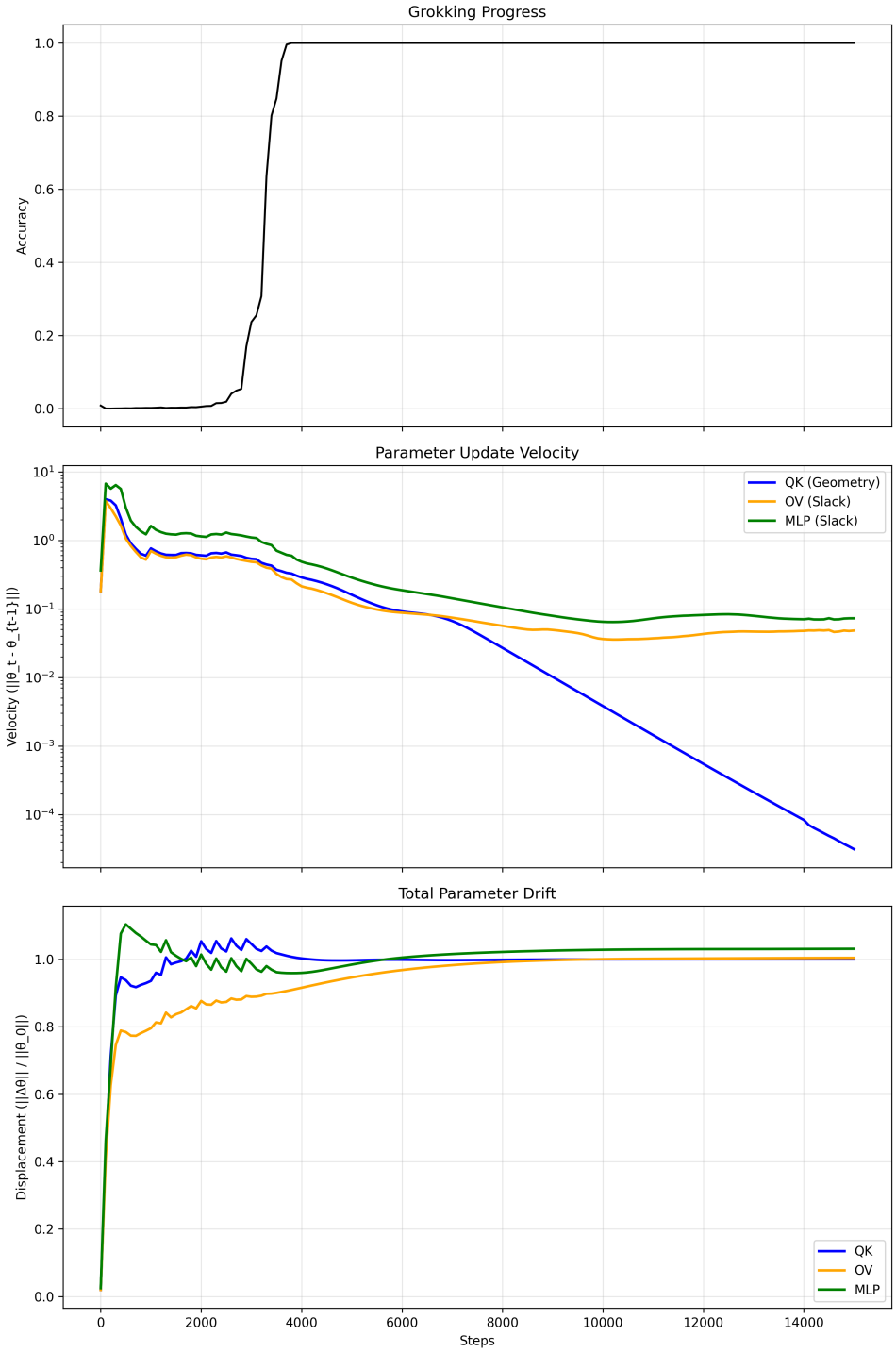

Figure 4: **Geometry Stabilizes Before Slack.** Parameter velocity tracking during model training. **Blue line (Geometry/QK):** The routing parameters stabilize early (Step ∼4000), dropping to negligible velocity. **Red line (Slack/OV+MLP):** The value and processing weights continue to evolve long after the geometry has stabilized. This supports the temporal ordering: G stabilizes first, defining the subspace, and S optimizes within it.

As shown in Figure 4, we observe a distinct phase structure that aligns with Nanda et al.'s [8] decomposition of grokking:

1. **Memorization / Plastic Phase (Steps 0–1400):** Both QK and OV/MLP circuits evolve rapidly. The model is fitting training data; no generalization yet. This corresponds to Nanda et al.'s "memorization phase."

2. **Circuit Formation / G Stabilization (Steps 1400–4000):** The QK circuit velocity drops sharply while OV/MLP velocity remains elevated. The routing structure is consolidating. Nanda et al. observed that the generalizing (Fourier) circuit forms during this phase, "well before grokking occurs."

3. **Cleanup / S Optimization (Steps 4000–15000):** QK velocity is near-zero; OV/MLP continues to adjust. The geometry is fixed; the model is optimizing slack within this fixed structure. Nanda et al.'s "cleanup phase" (where weight decay removes memorization components) occurs here.

4. **Post-Consolidation (Steps 15000+):** Both G and S have stabilized. The critical period has ended; Information Plasticity is lost.

This temporal ordering (G stabilizes before S) explains why mature models exhibit non-transferable structure: by the time training completes, the G-S coupling has been "baked in" through thousands of co-optimization steps.

7 Scaling Validation: 2-Layer Transformers

To verify our findings are not 1-layer artifacts, we extended all key experiments to 2-layer transformers ($d = 128$, 4 heads per layer). The scaling of our $G \times S$ framework to 2 layers is consistent with the observation that induction heads, which require 2-layer composition [9], exhibit similar circuit structure. The routing swap failure at 2 layers confirms that G-S coupling is not a shallow-model artifact.

7.1 Routing Swap (2-Layer)

We trained two 2-layer models (standard and noisy) to >99.9% accuracy. Swapping *all* QK parameters across *both* layers:

Model	**Accuracy**
Model A (Standard)	99.97%
Model B (Noisy)	99.99%
Hybrid (A + B's QK)	**0.0%**

Table 6: 2-Layer Routing Swap. Complete failure confirms G causality at depth.

The accuracy drop supports the interpretation that geometry causally determines behavior at 2 layers.

7.2 Sedation Test (2-Layer)

We clamped a converged 2-layer model to 60% of its natural amplitude and measured performance:

The margin-as-budget story holds: sedation doesn't affect clean accuracy but removes the stored noise buffer.

7.3 Early-Phase Subspace Probe (2-Layer)

For 2-layer models, we found that the warmup point matters significantly. Freezing QK at step 5000 (rather than step 2000 as in 1-layer) with 20000 training steps produced:

Condition	Clamp	Noise	Acc	Margin
Baseline Clean	Natural	0.0	100.0%	8.67
Baseline Noisy	Natural	2.0	99.6%	5.77
Sedated Clean	60%	0.0	100.0%	5.20
Sedated Noisy	60%	2.0	**73.1%**	1.20

Table 7: 2-Layer Sedation. Clean degradation: 0%, Noisy degradation: 26.5%.

Key Finding: 2-layer Early-Phase G requires capturing routing structure later in training. With appropriate hyperparameters, two high-accuracy solutions with orthogonal residual directions exist under frozen geometry (see Table 5, row 3–4), supporting the subspace interpretation at depth.

8 Ruling Out Alternative Mechanisms

To address potential counter-arguments regarding the "passive" nature of the system, we conducted a targeted deficit attribution experiment.

8.1 Experiment 4: Absence of Local Reflex (Attribution Stability)

A potential critique is that while *global* interaction (layer-wise margin) appears passive, the model might exhibit a *local* reflex where early attention layers redistribute probability mass to compensate for injury.

Protocol. We trained a 2-layer model on a complex Induction task (Repeat Sequence) to 100% generalization. We then injected acute noise ($\sigma = 2.0$) at Layer 0 and measured the shift in attention metrics at Layer 1.

Results (n=3 seeds):

- **High Entropy Shift ($d > 2.0$):** Noise significantly flattens the attention distribution (higher entropy).
- **Stable Routing Targets (Similarity > 0.85):** Despite the noise, the cosine similarity of the attention patterns (clean vs. injured) remains high.

Conclusion. The model does not reroute attention to "cleaner" tokens. The observed entropy increase is a *passive propagation of noise*, not an active compensatory mechanism. The high similarity scores suggest that the routing logic (Geometry) remains effectively fixed even under severe stress.

This experiment addresses the "self-repair" hypothesis [7]: perhaps models compensate for damage by rerouting through backup heads. We find no evidence of such rerouting at Layer 1 under Layer 0 injury; the attention mechanism is surprisingly rigid once trained.

8.2 Experiment 5: Forced Geometry Recovery (Transfer Failure)

A stronger test of G–S interdependence: can we transplant "Hardened" geometry onto a vulnerable baseline to *confer* robustness?

Protocol. We trained two models on Modular Arithmetic ($p = 113$, requiring grokking for generalization):

- **Baseline**: Clean training $\rightarrow$ 100% clean accuracy, $\sim$1% robust at $\sigma = 2.0$.
- **Noise-Trained**: Noise injection ($\sigma = 2.0$) during training $\rightarrow$ 93% clean accuracy, 90% robust at $\sigma = 2.0$.

We then created a **Hybrid** model by transplanting the Noise-Trained model's QK parameters (Geometry) onto the Baseline model's OV/MLP parameters (Suppressor).

Conclusion. The Hybrid model reduces to chance levels. Transplanting robust Geometry onto a non-robust Suppressor does *not* confer robustness. This provides evidence that:

1. **G is Necessary but Not Sufficient**: Routing alone cannot transfer robustness.
2. **G–S Coupling**: Geometry and Suppressor must be co-trained; they are bound together.
3. **Robustness is Emergent**: It arises from the joint training process, not modular component properties.

9 Emergent Stochasticity at Depth

At 2 layers, the $G \times S$ framework scales cleanly. At 4 layers, our observations suggest a qualitatively new phenomenon: **metastable oscillatory dynamics**.

While we note that the sample size is modest (5 seeds), the pattern is consistent across all runs. At 1–2 layers, the $G \times S$ framework and Nanda et al.'s [8] phase structure provide a clean, deterministic picture: geometry stabilizes, then slack optimizes, then the model converges. At 4 layers, this deterministic story breaks down. Training dynamics become *stochastic*: some seeds converge to stable solutions, others oscillate indefinitely, and the outcome cannot be predicted from early training behavior. This finding has implications for scaling: if stochasticity increases with depth, training outcomes at frontier scales may be fundamentally unpredictable without understanding the underlying landscape topology.

9.1 The Metastable Oscillatory Regime

Using the same early-phase freeze protocol (auto-detected critical period at ∼1500 steps, 90% validation accuracy threshold with hysteresis), 4-layer models exhibit:

1. **Transient Generalization** (Steps 1500–4000): Accuracy spikes to ∼100%, but the basin is shallow.
2. **Metastable Oscillations** (Steps 4000+): Models repeatedly collapse and recover (100% $\rightarrow$ 0.9% $\rightarrow$ 100% $\rightarrow$ 2%...). No monotonic convergence.
3. **Stochastic Escape**: Some seeds eventually lock in to stable attractors; others never escape the oscillatory regime.

Relation to Slingshot Mechanism. The oscillatory dynamics we observe (repeated collapse and recovery) resemble the "Slingshot Mechanism" identified by Thilak et al. [10], who observed cyclic phase transitions between stable and unstable training regimes. However, our analysis reveals crucial differences: (1) not all seeds escape the oscillatory regime, and (2) the outcome appears determined by landscape topology rather than optimization dynamics. This suggests either a stochastic variant of the mechanism or a related but distinct phenomenon where trajectories can be trapped in metastable basins indefinitely.

Metric	4-Layer	Interpretation
Stability Rate	40–50%	Stochastic, not deterministic
Mean Collapses	6.4 ± 3.1	High variance (stochastic fluctuations)
Time-to-Stability	∼19,500 steps	When achieved
Final Accuracy Std	44%	Signature of escape process

Table 8: **4-Layer Stability Characterization** (5 seeds). The huge variance (44% std on final accuracy) is consistent with stability being a stochastic escape process, not smooth convergence.

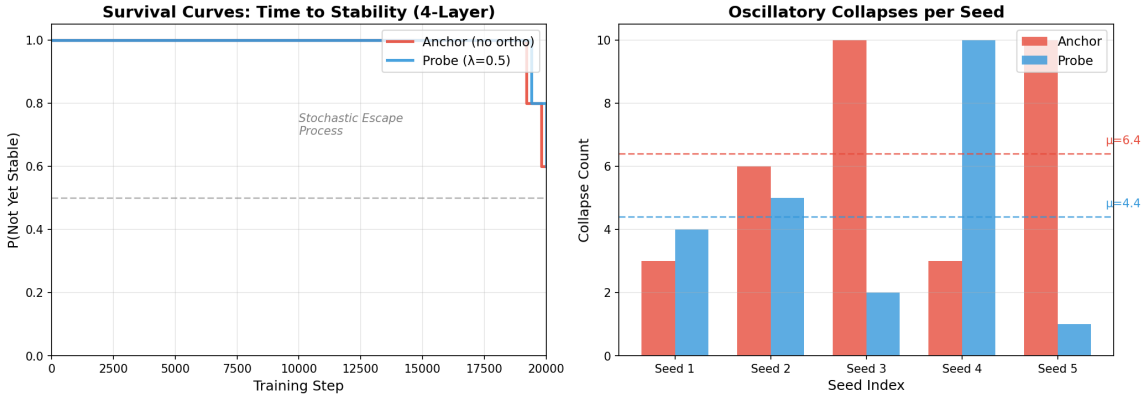

Figure 5: **Stochastic Escape at 4 Layers.** Left: Kaplan-Meier style survival curves showing time-to-stability. Both anchor (no ortho) and probe ($\lambda = 0.5$) exhibit step-function escape: models are "not yet stable" until they suddenly lock in. Right: Collapse counts vary widely across seeds (2–10), suggesting that collapses are stochastic fluctuations, not causally related to final stability.

9.2 Lambda Sweep: Falsifying Orthogonality as a Solution

A natural hypothesis is that regularization (e.g., orthogonality penalties) might help models escape the metastable regime. We tested this with $\lambda \in \{0, 0.05, 0.3\}$ (8 seeds per condition):

λ	Stability Rate	Mean Collapses	Mean Accuracy
0.0 (baseline)	**50%**	6.1 ± 2.7	0.594 ± 0.42
0.05	25%	6.0 ± 4.4	0.562 ± 0.37
0.3	25%	6.5 ± 5.0	0.490 ± 0.35

Table 9: **Lambda Sweep: Baseline Condition Achieves Highest Observed Stability.** The baseline condition ($\lambda = 0$) achieves the highest observed stability rate (50%), while adding orthogonality penalties *reduces* stability (25%) and degrades mean accuracy monotonically. (8 seeds per condition).

Key Finding: The baseline condition achieves the highest observed stability rate. The critical difference is that the Early-Phase subspace experiments (Section 6.4) *freeze* geometry, whereas the lambda sweep applies penalties during unconstrained training of all parameters. This suggests that orthogonal constraints on OV/MLP may interfere with the stochastic escape dynamics at depth when the geometry is also mobile. Collapse counts remain unchanged (∼6) across all λ, suggesting that the penalty does not affect the underlying landscape topology.

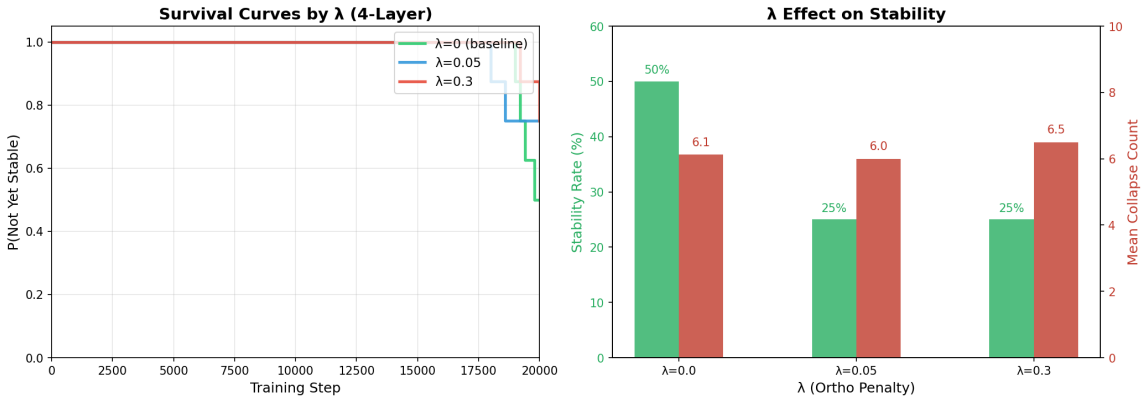

Figure 6: **Lambda Sweep Results.** Left: Survival curves by λ. The baseline ($\lambda = 0$, green) achieves the highest final stability rate (50%). Adding orthogonality penalties ($\lambda = 0.05$, $\lambda = 0.3$) *reduces* stability to 25%. Right: Summary statistics. Mean collapses remain unchanged ($\sim$6) while stability rate and accuracy degrade with increasing λ.

9.3 Disentangling Reference Coupling: 2×2 Factorial

A confound in the above experiments is that probe models train *alongside* an anchor, potentially gaining stabilizing effects from reference coupling rather than the orthogonality penalty itself. We ran a 2×2 factorial (6 seeds × 4 conditions) to disentangle these effects:

Condition	Reference	Ortho	Stability	Collapses
CE Only	×	×	50%	6.2 ± 2.9
CE + EMA-Self	×	×	50%	$\mathbf{1.8 \pm 2.3}$
Anchor (no penalty)	✓	×	**67%**	8.0 ± 2.7
Anchor + Ortho	✓	✓	50%	7.5 ± 1.5

Table 10: **2×2 Factorial: Reference vs. Ortho.** Main effects: Reference +8% stability, Ortho −8% stability. EMA-Self damps high-frequency oscillations without altering escape probability.

Main Effects:

- **Reference trajectory**: +8% stability (A0=50% vs A1=58%)
- **Ortho penalty**: −8% stability (B0=58% vs B1=50%)
- **EMA smoothing**: Reduces collapses 6.2 → 1.8, but does not improve escape probability

Interpretation. The anchor reference improves stability more than EMA-Self despite higher collapse counts, suggesting that weak directional coupling can bias trajectories toward stable basins, whereas EMA primarily smooths dynamics along existing trajectories. Neither intervention alters landscape topology; both act on training dynamics rather than the underlying representational constraints. This confirms that the earlier probe-vs-anchor results were partially confounded: the modest stabilizing effect came from reference coupling, not orthogonality. The ortho penalty consistently *destabilizes* training, likely via gradient interference.

9.4 Interpretation: Landscape Topology, Not Optimization

> *Stability at depth is not guaranteed by training objectives; it emerges probabilistically from a metastable regime. Loss term engineering does not reliably alter the underlying behavior.*

This insight explains:

- Why variance is so high at depth
- Why collapse happens even after successful grokking
- Why "scaling feels unpredictable"
- Why 'self-healing' narratives are misleading

The phenomenon is fundamentally about:

- **Basin depth**: Shallow attractors are easily escaped
- **Barrier height**: Unchanged by loss term tweaks
- **Global landscape topology**: Intrinsic to depth, not optimization details

This stands in sharp contrast to the deterministic dynamics observed in shallow models (1–2 layers), where the transition to generalization (grokking) is smooth and reliable. At depth (4 layers), the process becomes **stochastic**, defined by escape probability rather than convergence rate.

9.5 Serendipitous Validation: Training Continuation as Intervention

During development of our Othello-GPT baseline, a checkpoint resume bug inadvertently created a natural experiment. Our best checkpoint (88.75% legal move accuracy at step 10,000) was resumed with model weights intact but optimizer state discarded. Rather than maintaining performance, the model *gradually degraded* to 81.80% over 10,000 additional training steps—a slow drift rather than immediate collapse.

This observation has precedent. Gross et al. [5] model late-stage optimization as a stochastic dynamical system near a stationary distribution, not descent toward a point. Adam's momentum (m) and variance (v) estimates encode anisotropic noise structure—resetting them changes the effective diffusion tensor while leaving parameters unchanged. The gradual degradation we observed is precisely what this framing predicts: the noise covariance changed, but the model remained in-basin initially, then slowly drifted as the altered noise coupled differently to local curvature.

Controlled Validation. We tested four conditions on converged modular arithmetic models:

Condition	Description	1-Layer Result
CONTROL	Full state preserved	99.19% → 99.81%
RESET	Fresh optimizer, full LR	99.19% → 99.57% (±0.28%)
LOW_LR	Fresh optimizer, 0.1× LR	99.19% → 99.69%
WARMUP	Fresh optimizer, linear warmup	99.19% → 99.82%

Table 11: **Optimizer State Ablation (1-Layer).** At one layer, optimizer reset causes transient degradation with eventual recovery. Warmup and reduced LR both eliminate the effect.

Depth Reveals Metastability. At four layers, the dynamics changed qualitatively:

The critical finding: **CONTROL also collapsed catastrophically**. This falsifies the naive assumption that preserving optimizer state guarantees stability. The instability is not *caused* by optimizer reset—it is *revealed* by continued optimization at an inappropriate step size. This aligns with the Hydra effect [7]: deep networks form redundant pathways where small

Condition	Initial	Minimum	Final	Max Drop
CONTROL	99.5%	2.2%	98.2%	**97.3%**
RESET	63.8%	10.8%	99.6%	53.0%
WARMUP	99.5%	99.5%	99.6%	0.1%

Table 12: **Optimizer State Ablation (4-Layer).** CONTROL also collapsed catastrophically in 2 of 3 seeds, falsifying the assumption that preserving optimizer state guarantees stability.

perturbations can trigger large-scale reconfiguration. Here, temporal continuation itself acts as the intervention.

Mechanism: Hessian Spectral Geometry. The universal protection afforded by warmup is predicted by sharp-spectrum Hessian theory [6]. At depth, the loss landscape develops eigenvalue spikes—even "stable" minima are surrounded by steep walls in specific directions. A resumed run with full learning rate immediately samples these stiff modes. Warmup works by slowly reintroducing energy, allowing the system to re-equilibrate rather than slamming into high-curvature barriers.

Connection to $G \times S$ Framework. These results extend our understanding of metastability at depth. The same sharp landscape topology that makes 4-layer models sensitive to geometric perturbation also makes them vulnerable to step-size-induced instability. Both phenomena arise from the high-curvature loss landscape that emerges with depth—a property of landscape topology, not optimization dynamics.

What This Contributes Beyond Prior Work:

1. Treating optimizer state as part of the learned system, not scaffolding—empirically demonstrating that "noise geometry memory" becomes behaviorally load-bearing

2. Showing metastability under *continued training*, not explicit intervention—time-continuation as a perturbation class

3. Demonstrating that CONTROL can catastrophically fail, falsifying "just preserve state" assumptions

4. Reframing warmup as a stability boundary condition rather than optimization hygiene

Practical Implications. Transfer learning workflows that discard optimizer state are only safe because practitioners also use small learning rates and warmup—often without understanding why. For deep models, warmup is not merely beneficial but essential: it prevents the learning rate from coupling to high-curvature directions before the optimization trajectory has re-stabilized. This provides mechanistic grounding for what has been empirical folklore.

10 Limitations

Scale. Our experiments use 1–4 layer transformers on controlled tasks. Whether pre-allocated amplitude and the geometry–amplitude trade-off generalize to frontier-scale models remains an open question.

Amplitude definition. LayerNorm constrains scale up to learned gain, effectively imposing a fixed dynamic range at the decision site. We use signed logit margin as our primary amplitude metric to avoid the "constant by construction" critique.

Architecture. Our 2-layer validation shows the framework holds with depth. At 4 layers, we observe metastable dynamics that may reflect more complex $G \times S$ interactions, including cross-layer dependencies.

4-Layer Statistics. While our lambda sweep (8 seeds × 3 conditions) provides directional evidence, larger-scale studies would sharpen the stability rate estimates. The 50% vs. 25% difference (Fisher exact $p \approx 0.31$) is not individually significant. The qualitative phenomenon (stochastic escape vs deterministic convergence) is consistent across all seeds; the exact rates are secondary.

Relation to prior frameworks. Our G × S decomposition is complementary to, not a replacement for, existing frameworks:

- Nanda et al. [8] explain *what* algorithm grokked models learn (Fourier multiplication). We explain *how* that algorithm becomes structurally fixed (G stabilizes, then S optimizes within G).
- Achille et al. [1] describe the *phenomenology* of consolidation (Information Plasticity loss). We identify *which parameters* consolidate first (QK before OV/MLP).
- Thilak et al. [10] characterize the *dynamics* of late-stage training (Slingshot oscillations). We show that at depth, these dynamics become *stochastic* rather than deterministic.

11 Conclusion

Our experiments validate the decomposition $\Psi \sim f(G, S)$ where G (geometry/routing) and S (slack/amplitude) are separable but causally interdependent:

1. **No Gain Reflex:** Trained Transformers do not exhibit systemic amplitude up-regulation in response to acute damage. Robustness is not dynamically generated; it is spent from a fixed budget.
2. **Pre-allocated Amplitude:** High-precision tasks drive models to accumulate higher decision margins during training. This margin is structurally necessary; clamping it degrades performance under noise.
3. **Geometry vs. Amplitude Trade-off:** When robustness is explicitly enforced during training, models preferentially adapt geometry (reorganizing the representation basis) rather than increasing amplitude.
4. **G Causality:** Transplanting attention routing between converged models causes near-complete failure (99.99% → 0.02%), providing evidence that geometry causally constrains behavior. Slack learned under one geometry is incompatible with another.
5. **G→S Temporal Ordering:** Parameter drift analysis reveals that geometry stabilizes first. The **QK circuit** stabilizes on a fixed structure (Step ∼4000), while slack components (OV/MLP) continue to evolve. The model establishes geometric structure, then optimizes within it.
6. **G Defines a Subspace:** Under "Early-Phase Geometry" (frozen before generalization), two orthogonal slack allocations both achieve 100% accuracy. Geometry defines a *solution subspace*; training specializes the model into one specific path within it.
7. **Depth-Dependent Metastability:** At 4 layers, a new phenomenon emerges: stability is not deterministic but stochastic. Models oscillate through collapse/recovery cycles before escaping to stable attractors. Neither loss-term engineering nor dynamical smoothing reliably improves escape probability, suggesting that the phenomenon reflects intrinsic **landscape topology**, not optimization dynamics.

The picture that emerges is one of **progressive specialization with depth-dependent dynamics**. At shallow depth (1–2 layers), geometry stabilizes early, defining a broad subspace; the model stabilizes on a specific slack configuration deterministically. At greater depth (4 layers), this process becomes *stochastic*, where the model may not escape metastable oscillatory regimes, and stability is not guaranteed by training objectives.

What appears as "compensation" in neural networks reflects either pre-allocated slack or dynamical smoothing; neither constitutes real-time structural adaptation. The learned structure is non-transferable. And at depth: stability is probabilistic, not deterministic.

Hypotheses. Based on these findings, we propose the following testable hypotheses: (1) Freezing QK circuits during the critical period [1] in large-scale models will preserve capability but prevent substantial behavioral modification through fine-tuning, since the geometry is already consolidated; (2) Robustness fine-tuning applied after circuit formation [8] will have limited effect unless the QK circuit retains plasticity—the geometry must be modifiable for the model to learn robust representations; (3) Increasing depth will monotonically increase the variance of training outcomes at fixed width, independent of dataset size; and (4) The Slingshot-like oscillations [10] observed at 4 layers will intensify with depth, making training outcomes at frontier scales increasingly stochastic and dependent on random seed.

Implications. These structural constraints explain why late-stage fine-tuning often fails to alter fundamental model behavior (the geometry has already stabilized), why robustness rarely transfers between models (slack is bound to specific trajectories), and why scaling behaviors can feel unpredictable at depth (the underlying dynamics are metastable, not smooth). Understanding these progressively hardening structures is essential for designing controllable AI systems.

References

[1] Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep networks. In *International Conference on Learning Representations*, 2019.

[2] Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. Mma training: Direct input space margin maximization through adversarial training. 2020.

[3] Nelson Elhage, Neel Nanda, Catherine Olsson, et al. A mathematical framework for transformer circuits. In *Transformer Circuits Thread*, 2021.

[4] Gamaleldin Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. Large margin deep networks for classification. In *NeurIPS*, 2018.

[5] David J Gross, Neel Nanda, et al. Weight fluctuations in deep linear neural networks and a derivation of the inverse-variance flatness relation. *arXiv preprint arXiv:2403.00322*, 2024.

[6] Zhenyu Liao and Michael W Mahoney. Hessian eigenspectra of more realistic nonlinear models. In *NeurIPS*, 2021.

[7] Thomas McGrath et al. The hydra effect: Emergent self-repair in language model computations. *arXiv preprint*, 2023.

[8] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations*, 2023.

[9] Catherine Olsson, Nelson Elhage, Neel Nanda, et al. In-context learning and induction heads. In *Transformer Circuits Thread*, 2022.

[10] Vimal Thilak, Etai Littwin, Shuangfei Zhai, Omid Saremi, Roni Paiss, and Joshua Susskind. The slingshot mechanism: An empirical study of adaptive optimizers and the grokking phenomenon. *arXiv preprint arXiv:2206.04817*, 2022.

[11] Kevin Wang et al. Interpretability in the wild: A circuit for indirect object identification in gpt-2 small. In *ICLR*, 2023.