
The Readout Jacobian Reveals Grokking Representations: A Layerwise eNTK View

Maximilian Nicholson
University of Bath
mn866@bath.ac.uk

Abstract

Where and when does a learned algorithm become visible in a network’s parameter-gradient geometry? Mechanistic analyses of modular addition locate the trigonometric computation in the MLP, so one might expect the MLP empirical neural tangent kernel (eNTK) to show the clearest task structure. After independently normalizing each parameter block, we find the opposite: in a 20-seed, $p = 113$ experiment, final unembedding alignment completely separates seeds grokked by a 7,000 epoch landmark from right-censored seeds (with Mann-Whitney $p = 0.0002$, $r_{\text{rb}} = 1.00$), tracking final accuracy ($\rho = 0.928$). A prospectively locked GELU with pre-LayerNorm replication reproduces this normalized result. The apparent contradiction dissolves through the readout identity derived by Lin [2025]: $K_t^{(\text{unembed})} = pG_t$ in our scalarized convention, making the normalized readout kernel exactly the geometry of the final residual representation. That visibility does not imply universal additive dominance. The unembedding holds a median 90.8% of endpoint target mass in the primary ReLU/no-LayerNorm model, whereas the GELU+pre-LayerNorm condition assigns median shares of 66.9% to embeddings, 28.1% to attention, and 2.85% to the unembedding. Activation Grams then reveal where the visible representation arises: addition alignment is negligible through attention, rises to 0.524 at the MLP output, and remains high downstream. Finally, in the ten grokked primary models, immediate removal of a rank-5 eNTK-guided sum-orbit subspace reduces mean accuracy from 0.9987 to about 0.10, while retaining only it preserves about 0.25. This supports necessity for these evaluated solutions and partial immediate sufficiency, not unique mechanistic ownership. The readout is therefore a window onto an upstream representation, while visibility, additive attribution, pathway emergence, and functional relevance remain distinct.

Keywords. grokking, empirical neural tangent kernel, mechanistic interpretability, neural artifacts, optimization trajectories, representation geometry

1 Introduction

Grokking transforms a normally hidden process—the acquisition of a generalizing algorithm—into an observable trajectory: a model first memorizes its training set and only later generalizes to unseen examples [Power et al., 2022]. Modular addition thus allows us to ask not only whether a task representation has formed, but also where it becomes visible in the network and in the geometry induced by its parameter gradients.

The mechanistic circuit suggests an immediate answer. In the one-layer Transformer studied by Nanda et al. [2023], embeddings supply token frequencies, attention combines the inputs, and the MLP computes trigonometric identities that render $a + b \pmod{p}$ linearly decodable. If a layerwise

eNTK localized the computation itself, the MLP should therefore show the strongest sum-Fourier structure. The feature-learning view points in the same direction: finite networks that leave the lazy regime reorganize their parameter-gradient geometry around task structure [Jacot et al., 2018, Chizat et al., 2019, Atanasov et al., 2021, Shan and Bordelon, 2022], and grokking is often described as a lazy-to-rich transition [Kumar et al., 2024, Mohamadi et al., 2024].

Our measurements present a puzzle. In the primary 20-seed, $p = 113$ experiment, the MLP block exhibits little change on an independently normalized sum-orbit alignment scale. Instead, the unembedding develops the strongest alignment, distinguishes the landmark groups, and tracks accuracy across seeds. Of the summed positive normalized block-shape changes, 95.8% correspond to the unembedding and 0.0% to the MLP. Taken at face value, this result would move the modular-addition algorithm from the MLP to the readout, contradicting the circuit picture.

The conflict disappears once localization is split into four questions. Independently normalized kernels ask where task geometry is most visible; a shared-denominator decomposition asks which blocks contribute additive kernel mass; activation measurements ask where the geometry emerges in the *forward pathway*; and interventions test functional importance under a specified perturbation. The unembedding makes these distinctions unusually transparent. Lin [2025] derives its kernel in per-class form as a class-diagonal residual Gram matrix. Summing over output classes in our scalarized convention gives

$$K_t^{(\text{unembed})}(x, x') = p \cdot \langle h_T(x), h_T(x') \rangle.$$

After centering and trace-normalization, the unembedding eNTK is therefore exactly the normalized Gram matrix of the final residual stream. Its alignment shows that the representation handed to the decoder has become strongly dependent on the sum coordinate over the probed orbits. It does not, on its own, identify particular Fourier frequencies or show that the readout constructed that representation.

The experiments trace that distinction from observation to mechanism. Landmark, dose-response, and modulus analyses first establish normalized readout geometry as a robust diagnostic of generalization. Matched additive decompositions then reveal its boundary: the unembedding dominates both normalized and additive change in the primary ReLU/no-LayerNorm condition, but the locked GELU+pre-LayerNorm landmark places additive growth in embeddings and attention instead. Activation Grams next trace the representation backward to a sharp transition at the MLP output, and eNTK-guided interventions show that the rank-5 sum-orbit subspace is necessary for the current computation of the ten evaluated grokked primary models and partially sufficient for their immediate performance, without implying unique mechanistic ownership. The emerging story is not that the readout replaces the MLP as the site of computation, but that its Jacobian exposes an upstream representation with particular clarity.

Relation to Lin (2025). Lin [2025] establish Fourier alignment in layerwise eNTKs during grokking and derive the readout residual-Gram identity. Our contribution is not that identity itself. We test whether the resulting readout localization holds up across seeds, moduli, and a prospectively locked architectural replication; distinguish normalized representation visibility from additive kernel attribution; locate where in the forward pathway the representation emerges; and directly intervene on the eNTK-selected activation subspace.

Contributions.

1. **Task-specific blockwise metrics.** We define modular-addition alignment and symmetry metrics over a transformation-closed probe set, distinguishing within-block shape comparison from additive attribution.
2. **Robust representation visibility.** Across the primary landmark, five additional moduli, and a locked GELU+pre-LayerNorm replication, unembedding alignment gives the clearest normalized eNTK signature of generalization.
3. **Condition-dependent additive attribution.** Matched within-condition decompositions show that normalized readout dominance does not determine additive localization: the unembedding dominates additive growth at the ReLU/no-LayerNorm endpoint, whereas embeddings and attention dominate at the GELU+pre-LayerNorm locked landmark.

4. **Pathway and intervention evidence.** Building on Lin’s readout identity, activation Grams locate the transition at the MLP output, while rank-5 interventions support necessity for the ten evaluated grokked primary models and partial sufficiency for their immediate performance, without assigning unique ownership.

2 Related Work

Grokking and mechanistic circuits. Power et al. [2022] introduced grokking; Nanda et al. [2023] subsequently reverse-engineered a Fourier-based modular-addition circuit and proposed progress measures for its formation, while Gromov [2023] derived features for broader modular-arithmetic tasks. Other analyses characterize grokking as a transition out of kernel-like dynamics [Kumar et al., 2024, Mohamadi et al., 2024]. These results motivate asking how the circuit manifests in parameter-gradient geometry.

NTK and feature learning. The NTK characterizes lazy training [Jacot et al., 2018, Yang, 2020, Lee et al., 2019, Arora et al., 2019], whereas deviations from a fixed kernel signal feature learning [Chizat et al., 2019, Geiger et al., 2020, Atanasov et al., 2021, Shan and Bordelon, 2022]. Spectral studies examine frequency bias [Rahaman et al., 2019, Tancik et al., 2020] and task–model alignment in kernel regression [Canatar et al., 2021]. This suggests that an evolving eNTK may reveal a learned algorithm, but not which block reveals it most clearly.

Empirical NTK analysis. Finite-width estimators make time-resolved eNTK analysis tractable [Novak et al., 2022, Mohamadi et al., 2023]. Most directly, Lin [2025] show Fourier alignment in layer-wise modular-addition eNTKs and derive the unembedding kernel as a class-diagonal residual Gram. We use this identity to link apparent readout localization to the upstream circuit, then examine this connection across seeds, architectures, activations, controls, and activation-subspace interventions.

3 Setup

To trace the representation from parameter geometry to forward computation, we keep the task and block partition fixed wherever possible and vary only one interpretive question at a time.

Task and architecture. The primary experiment trains a one-layer Transformer [Vaswani et al., 2017] on modular addition with $p = 113$ and train fraction $\rho = 0.3$. The model has 4 heads, $d_{\text{model}} = 128$, $d_{\text{mlp}} = 512$, a ReLU MLP [Nair and Hinton, 2010], and no LayerNorm [Ba et al., 2016]. Each of the 20 random seeds is trained full-batch for 7,000 epochs with AdamW [Loshchilov and Hutter, 2019] at learning rate 10^{-3} and weight decay 1.0.

Since the argument depends on a precise block decomposition, we spell out the forward pass explicitly. For the sequence $x = (a, b, =)$, let $X_0 \in \mathbb{R}^{3 \times d}$ denote the sum of token and positional embeddings. For attention head h ,

$$Q_h = \mathcal{N}_{\text{attn}}(X_0)W_Q^{(h)}, \quad K_h = \mathcal{N}_{\text{attn}}(X_0)W_K^{(h)}, \quad V_h = \mathcal{N}_{\text{attn}}(X_0)W_V^{(h)},$$

$$A_h = \text{softmax}\left(Q_h K_h^\top / \sqrt{d_h} + M_{\text{causal}}\right), \quad X_1 = X_0 + \sum_h A_h V_h W_O^{(h)},$$

and

$$U = \phi(\mathcal{N}_{\text{mlp}}(X_1)W_{\text{in}} + b_{\text{in}}), \quad X_2 = X_1 + U W_{\text{out}} + b_{\text{out}}, \quad f(x) = X_2(x)_{=,:}^\top W_U.$$

The primary and modulus runs use identity normalizations and ReLU; the locked replication uses pre-LayerNorm and GELU. Parameters are partitioned disjointly into embedding (W_E, W_{pos}), attention (W_Q, W_K, W_V, W_O), MLP ($W_{\text{in}}, b_{\text{in}}, W_{\text{out}}, b_{\text{out}}$), and unembedding (W_U) blocks. In the pre-LayerNorm model, the pre-attention gain/bias belong to the attention block, and the pre-MLP gain/bias belong to the MLP block. This convention keeps the same four-way question consistent across both architectures: where does task geometry become visible?

Why 7,000 epochs. Test accuracy is evaluated every 50 epochs, and t_{grok} is the first evaluation in a run of two consecutive checkpoints above 95% accuracy. The exploratory epoch-7,000 horizon was chosen after inspecting the baseline trajectories because it gives an approximately balanced 10/10 snapshot. Ungroked seeds are right-censored, not a permanent population, so the landmark p -values are descriptive and selection-conditional. We complement them with continuous across-seed analyses and the prospectively locked GELU confirmation.

eNTK computation. For block ℓ , let $J_t^{(\ell)}(x) = \partial f(x; \theta_t) / \partial \theta^{(\ell)} \in \mathbb{R}^{p \times d_\ell}$. Its exact scalarized eNTK is

$$K_t^{(\ell)}(x, x') = \langle J_t^{(\ell)}(x), J_t^{(\ell)}(x') \rangle_F = \sum_{c=1}^p \langle \nabla_{\theta^{(\ell)}} f_c(x), \nabla_{\theta^{(\ell)}} f_c(x') \rangle.$$

We estimate it using random output projections [Novak et al., 2022]. For independent Rademacher vectors $z_r \in \{-1, +1\}^p$, define

$$g_r(x) = \frac{z_r^\top f(x)}{\sqrt{R}}, \quad K_{t,R}^{\text{RP},(\ell)}(x, x') = \sum_{r=1}^R \langle \nabla_{\theta^{(\ell)}} g_r(x), \nabla_{\theta^{(\ell)}} g_r(x') \rangle.$$

We use the same z_r for all examples and blocks at a checkpoint, $R = 16$, probe chunks of 64, and checkpoints every 500 epochs for the ReLU runs and every 250 epochs for the GELU replication. Unless otherwise stated, final values average the last four eNTK checkpoints.

Proposition 1 (Estimator properties). *For the disjoint block partition,*

$$K_t = \sum_{\ell} K_t^{(\ell)}, \quad K_{t,R}^{\text{RP}} = \sum_{\ell} K_{t,R}^{\text{RP},(\ell)}, \quad \mathbb{E} K_{t,R}^{\text{RP},(\ell)} = K_t^{(\ell)}.$$

Every realized probe matrix is positive semidefinite. For the bias-free unembedding, the projected estimator is exact for every realization and every $R \geq 1$:

$$K_{t,R}^{\text{RP},(\text{unembed})}(x, x') = p \langle h_T(x), h_T(x') \rangle.$$

Proof. Direct-sum concatenation of block gradients proves both additivity statements. Suppressing t, ℓ , the chain rule gives $\nabla g_r(x) = R^{-1/2} J(x)^\top z_r$; using $\mathbb{E}[z_r z_r^\top] = I_p$ proves unbiasedness, while stacking the projected gradients writes every realization as a Gram matrix and proves positive semidefiniteness. For the unembedding, $\nabla_{W_U} g_r = R^{-1/2} h_T z_r^\top$ and $\|z_r\|^2 = p$, which gives exactness after summing over r . Section 5 derives and interprets the resulting readout identity in detail. $\square$

Task-specific probes and projectors. To test for the relevant geometry, the probe must preserve the task’s symmetry rather than sample unrelated examples. Write $q_+(a, b) = a + b$ for addition and $q_-(a, b) = a - b$ for subtraction. For $q_{\alpha, \beta}(a, b) = \alpha a + \beta b$ and orbit value τ , the complete orbit is enumerated by

$$\mathcal{O}_\tau^q = \{(a, \beta^{-1}(\tau - \alpha a)) : a \in \mathbb{Z}_p\}.$$

Both tasks use the six orbit values $\Omega = \{0, 17, 30, 52, 87, 88\} \pmod{p}$, giving $S_q = \bigcup_{\tau \in \Omega} \mathcal{O}_\tau^q$ and $n = 6p$. The map $T_\delta^q(a, b) = (a + \beta\delta, b - \alpha\delta)$ preserves q , so the probe is closed under its task symmetry. Let Z_q be the incidence matrix of the observed q values. The orbit-constant and centered projectors are

$$P_q = Z_q(Z_q^\top Z_q)^{-1} Z_q^\top, \quad P_q^0 = H P_q H, \quad H = I - n^{-1} \mathbf{1} \mathbf{1}^\top. \quad (1)$$

On six complete task orbits, $\text{rank}(P_q) = 6$ and $\text{rank}(P_q^0) = 5$, and the restricted characters $e^{2\pi i \omega q/p}$ span the orbit-constant space. The mismatched-coordinate projector has rank p because each complete orbit contains every value of the other coordinate when p is odd.

For each four-block checkpoint, we evaluate the ordered probe, draw one fixed Rademacher matrix, differentiate every projected output, flatten gradients within each block, accumulate block Gram matrices, sum raw blocks, symmetrize via $(K + K^\top)/2$, center, and then apply either independent or shared-denominator normalization. Directions below 10^{-10} of the largest retained singular/eigenvalue are treated as numerical zero.

Beyond the primary run, the four-block analysis covers 20 seeds at each $p \in \{59, 83, 89, 101, 127\}$, using empirically selected, median-targeting horizons 15,000, 8,500, 10,000, 7,500, and 5,500; these are exploratory scope checks. The prospectively locked 20-seed GELU/pre-LayerNorm replication at $p = 113$ fixed its architecture and epoch-2,250 landmark before evaluating new seeds. A locked 20-seed subtraction control failed to generalize and therefore serves only as a negative scope check. All use the same six-orbit construction and $R = 16$; selected modulus checkpoints use $R \in \{32, 64\}$, and the GELU analysis includes 50 permutation nulls per seed.

Layerwise decomposition. The first localization question concerns visibility rather than magnitude. Although the raw eNTK decomposes additively as $K_t = \sum_\ell K_t^{(\ell)}$, we independently center and trace-normalize each block, $\hat{K}_t^{(\ell)} = HK_t^{(\ell)}H/\text{Tr}(HK_t^{(\ell)}H)$, before comparing its shape with the task subspace. This deliberately removes block magnitude and therefore destroys additivity: the resulting scores rank normalized geometries rather than contributions to the full eNTK. The exact relation to the normalized full kernel is

$$\tilde{K}_t = \sum_\ell w_t^{(\ell)} \hat{K}_t^{(\ell)}, \quad w_t^{(\ell)} = \frac{\text{Tr}(HK_t^{(\ell)}H)}{\text{Tr}(HK_tH)}.$$

Indeed, writing $d_t^{(\ell)} = \text{Tr}(HK_t^{(\ell)}H)$ and $D_t = \text{Tr}(HK_tH)$, Proposition 1 gives $HK_tH = \sum_\ell HK_t^{(\ell)}H$ and hence $D_t = \sum_\ell d_t^{(\ell)}$. Dividing by D_t and substituting $HK_t^{(\ell)}H = d_t^{(\ell)} \hat{K}_t^{(\ell)}$ proves the identity and $\sum_\ell w_t^{(\ell)} = 1$. The weights are exactly what the independent normalization discards, and they motivate the second localization question.

Matched additive target-mass change. To determine which blocks contribute task-aligned mass, we restore the scale removed above. Let P_{add}^0 denote the projector onto the five nonconstant centered sum modes. We retain block magnitude while preserving additivity through the shared full-kernel denominator

$$K_t^{\star(\ell)} = \frac{HK_t^{(\ell)}H}{\text{Tr}(HK_tH)} \quad \text{and} \quad C_t^{(\ell)} = \text{Tr}(P_{\text{add}}^0 K_t^{\star(\ell)}).$$

The corresponding endpoint share is $s_t^{(\ell)} = C_t^{(\ell)} / \sum_j C_t^{(j)}$. Each $K_t^{\star(\ell)}$ is positive semidefinite, so $C_t^{(\ell)} = \text{Tr}(P_{\text{add}}^0 K_t^{\star(\ell)} P_{\text{add}}^0) \geq 0$; linearity gives $\sum_\ell C_t^{(\ell)} = \text{Tr}(P_{\text{add}}^0 \tilde{K}_t)$ and therefore $\sum_\ell s_t^{(\ell)} = 1$ whenever the denominator is positive. Endpoint differences may nevertheless be signed. We apply the same construction at the checkpoints used for each normalized analysis:

$$\Delta C_{\text{ReLU}}^{(\ell)} = C_{7000}^{(\ell)} - C_0^{(\ell)}, \quad \Delta C_{\text{GELU}}^{(\ell)} = C_{2250}^{(\ell)} - C_0^{(\ell)}.$$

The latter is a baseline-to-locked-landmark change; ten seeds remain right-censored at epoch 2,250. We report seed-bootstrap 95% confidence intervals and paired two-sided Wilcoxon signed-rank tests with Holm correction across all six block pairs.

Symmetry metrics. Our primary score asks whether the leading kernel geometry lies in the task-relevant orbit-constant space. The projector P_{add} from Eq. 1 is equivalently the saturated span of the restricted sum-Fourier characters, so it measures dependence on the six probed sums without identifying particular frequencies. One direction is constant, and centered kernels annihilate it. With $Q_{m,t}$ the top- m eigenprojector of $\hat{K}_t^{(\ell)}$ and $m \in \{5, 6\}$, we report

$$\bar{A}_{m,t}^{\text{add}}(\ell) = \frac{\frac{1}{m} \text{Tr}(P_{\text{add}} Q_{m,t}) - r/n}{\min(m, r)/m - r/n}. \quad (2)$$

This affine reporting convention is neither additive nor unit-normalized after centering: because centering removes the constant direction, exact recovery with $m = r = 6$ has a ceiling of approximately 0.832 at $p = 113$, while $m = 5$ directly targets the informative centered subspace. As boundary checks, R^{sum} measures relative deviation under diagonal translations $(a, b) \mapsto (a + \delta, b - \delta)$, and E^{stat} measures the relative residual after projection onto matrices constant in the sum difference $q_+(x) - q_+(x')$. The two diagnostics are complementary rather than equivalent.

Activation pathway and task-subspace interventions. The kernel analyses reveal where task geometry is visible, but not where it appears in the forward pass or whether it matters to the computation. We address those questions at the epoch-7,000 primary checkpoints by applying the same centered sum-orbit metric to activation Gram matrices at the attention output, post-attention residual, MLP output, final residual, and logits. Let A_T contain the final-residual probe activations and let P_{add}^0 be the centered rank-5 task projector. For each grokked seed we construct the guided activation basis by

$$A_{\text{task}} = P_{\text{add}}^0 H A_T = U \Sigma V^\top, \quad B = V_{:,1:5}.$$

Because $K_U = p A_T A_T^\top$ and $P_{\text{add}}^0 H = H P_{\text{add}}^0 = P_{\text{add}}^0$,

$$P_{\text{add}}^0 H K_U H P_{\text{add}}^0 = p (P_{\text{add}}^0 H A_T) (P_{\text{add}}^0 H A_T)^\top = p U \Sigma^2 U^\top.$$

Moreover, for each nonzero singular triplet, $v_j = \sigma_j^{-1} A_T^\top H P_{\text{add}}^0 u_j$. Thus U contains the task-restricted readout-eNTK modes and B is their activation-space lift. All ten evaluated seeds have five retained singular values above 10^{-10} of the largest. At both the MLP output and final residual, guided removal uses $h(I - B B^\top)$ and retention uses $h B B^\top$; outcomes are measured immediately, without parameter updates, using accuracy, loss, and correct-logit margin.

A random rank-matched control uses the orthonormal factor B_0 obtained by reduced QR decomposition of a CPU-generated Gaussian $d \times 5$ matrix. For model seed s , stage indicator $j = 1\{\text{final residual}\}$, and replicate $r \in \{0, \dots, 19\}$, the basis seed is $700000 + 10000s + 100j + r$. The same basis is reused across removal versus retention and rank-matched versus rank-and-norm-matched controls. For probe activations A_s , the latter uses $\gamma_s = \|A_s B B^\top\|_F / \|A_s B_0 B_0^\top\|_F$ and replaces the guided component by $\gamma_s h B_0 B_0^\top$. Thus matching is on the Frobenius norm of the projected activation component, not the basis norm. Each Holm adjustment covers the two guided-versus-control contrasts for one stage, intervention mode, and outcome, not all outcomes and stages globally. The archive includes all 1,640 draw-level rows and a seed manifest; the original sampled basis tensors and PyTorch version were not retained, so bitwise regeneration is not claimed.

Reproducibility artifact. The artifact at <https://anonymous.4open.science/r/entk-grokking-5F13> contains configurations, deterministic seed rules, draw-level random controls, results, checksums, and reproduction instructions. Runs use one GPU per process; original hardware included NVIDIA RTX 6000 Pro and H100 GPUs, and an A100 reproduction is recorded. Reconstructed per-seed costs are 68 seconds for the 5,000-epoch GELU+pre-LayerNorm run with 21 eNTK checkpoints (plausible range 55–85 seconds) and 46 seconds for the 7,000-epoch ReLU/no-LayerNorm run with two checkpoints (35–60 seconds). Estimated peak memory is 2.0 GB allocated (1.5–3.0 GB) and 3.0 GB reserved (2.0–5.0 GB). The archived-and-reproduction workload is approximately 1.4 GPU-hours (1.1–1.9), and the full project including exploratory, failed, and repeated runs is approximately 2.5 GPU-hours (2.0–4.0); Appendix Table 4 gives the estimation basis and device-specific timings.

4 Results

The results align with the localization puzzle introduced above. We first determine where task geometry is most apparent, then assess whether additive attribution provides consistent answers across the two tested condition–landmark pairs, and finally trace and intervene in the representation revealed by the readout kernel.

4.1 A localization puzzle at the exploratory landmark

The exploratory epoch-7,000 landmark catches the baseline midway through its transition: 10 seeds have crossed the criterion and 10 remain right-censored with accuracies between 0.050 and 0.929; an independent implementation reproduces the partition. By epoch 15,000, all 20 seeds grok (median $t_{\text{grok}} = 7,075$), confirming that the split is a temporary snapshot rather than two stable populations.

Evidence status. **Exploratory:** ReLU/no-LayerNorm and modulus-specific landmark p -values describe separation at empirically chosen snapshots. **Prospectively locked confirmation:** the GELU+pre-LayerNorm architecture, epoch-2,250 landmark, metrics, and analysis were fixed before evaluating 20 new seeds and are reported separately below.

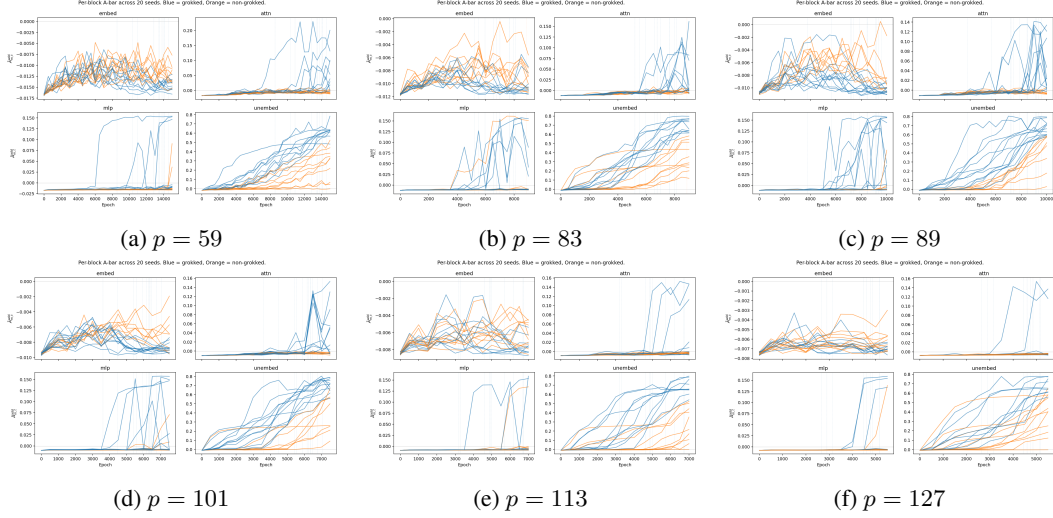


Figure 1: **Layerwise $\bar{A}^{\text{add}}$ trajectories across all completed moduli.** Each panel shows embedding, attention, MLP, and unembedding trajectories; blue indicates seeds that grokked within the modulus-specific budget, while orange indicates right-censored seeds that did not grok within the budget. The unembedding exhibits the largest positive alignment change in every completed modulus run.

Against this evolving backdrop, the blockwise comparison produces the central surprise. Figure 1 shows little separation in embedding or MLP geometry, but a pronounced divergence in the unembedding trajectories. At the landmark, unembedding alignment ranges from 0.462 to 0.720 among seeds grokked by budget (median 0.645), compared with -0.001 to 0.446 among right-censored seeds (median 0.151). This separation is complete (Mann–Whitney $p = 0.0002$, rank-biserial $r_{\text{rb}} = 1.00$), whereas the upstream effects are much weaker: attention gives $p = 0.1405$, $r_{\text{rb}} = 0.40$, the MLP gives $p = 0.4727$, $r_{\text{rb}} = 0.20$, and embedding alignment remains near zero. Complete per-seed values are included in the artifact.

The circuit and the kernel thus appear to indicate different aspects: the MLP is involved in constructing the Fourier features, while the readout Jacobian most directly shows whether those features aid generalization. The remainder of the analysis investigates whether this represents a contradiction or simply a difference in the interpretation of the two measurements.

4.2 The surprise is a change in normalized shape

The endpoint separation is part of a broader shift in normalized shape. Of the summed positive baseline-to-budget changes in $\bar{A}^{\text{add}}$, the unembedding accounts for 95.8%, compared with 3.8% for attention, 0.4% for embeddings, and 0.0% for the MLP. This quantifies the visual impression but also exposes the limit of the result. Because each block is normalized independently and the score depends nonlinearly on a leading eigenspace, the ratio answers, “whose normalized geometry changes most?” It does not answer, “which block contributes most to the full kernel?”

4.3 The readout signal tracks the degree of generalization

Before changing the attribution question, we ask whether the landmark has merely converted a continuous trend into two convenient groups. It has not. Across all 20 seeds, final unembedding alignment rises monotonically with final test accuracy (Spearman $\rho = 0.928$, $p = 3.76 \times 10^{-9}$; Table 1). The MLP association is smaller ($\rho = 0.479$, $p = 0.0325$) and spans a much narrower range. The readout signal therefore tracks partial progress toward generalization rather than simply reproducing the 10/10 label. Because alignment and accuracy are measured at the same endpoint, however, the relationship remains diagnostic rather than predictive or causal.

Table 1: **Exploratory cross-modulus summary of the unembedding diagnostic.** GB/NGB represent median final alignments for seeds grokked/not grokked by the empirically chosen modulus-specific budget. MW denotes the descriptive landmark Mann–Whitney comparison and r_{rb} its rank-biserial effect size. The ratio measures the unembedding share of summed positive independently normalized block changes, not additive full-kernel attribution.

p	grokked/20	GB med.	NGB med.	MW p	r_{rb}	unemb. ratio	Spearman $\rho(p)$
59	10	0.603	0.273	0.0002	1.00	69.0%	0.868 (6.94×10^{-7})
83	10	0.719	0.251	0.0002	1.00	72.0%	0.902 (5.61×10^{-8})
89	12	0.642	0.394	1.59×10^{-5}	1.00	68.5%	0.926 (4.94×10^{-9})
101	12	0.686	0.311	1.59×10^{-5}	1.00	72.4%	0.802 (2.14×10^{-5})
113	10	0.645	0.151	0.0002	1.00	95.8%	0.928 (3.76×10^{-9})
127	9	0.532	0.286	0.0008	0.90	99.4%	0.892 (1.31×10^{-7})

4.4 The pattern persists, with clear boundaries

The first possible explanation is methodological: perhaps the result depends on the chosen rank or on projection noise. Neither accounts for it. Restricting the analysis to the five informative centered modes strengthens the primary separation, with unembedding medians of 0.752 for seeds grokked by budget and 0.166 for right-censored seeds (Mann–Whitney $p = 0.0002$). Increasing the projection count from $R = 16$ to 32 or 64 leaves the $p = 113$ unembedding median at 0.647 to three decimals.

Nor is the phenomenon confined to $p = 113$. Across $p \in \{59, 83, 89, 101, 127\}$, the unembedding again has the largest positive-change ratio (68.5%–99.4%), and its endpoint correlation with accuracy ranges from 0.802 to 0.926 (Table 1). Because the modulus-specific horizons were empirical median-targeting choices, these runs broaden the observed scope without constituting pooled or independently selected confirmations.

Replication across moduli does not make the claim universal. Alternative symmetry metrics are less uniform: the unembedding leads both R^{sum} and E^{stat} at $p = 59, 83, 89, 101$, the metrics disagree at $p = 113$, and neither favors the unembedding at $p = 127$. The recurring result is therefore specifically unembedding-dominant change in the normalized alignment score, not agreement among every conceivable symmetry diagnostic.

The locked subtraction control is a negative boundary, not a successful replication. None of its 20 seeds grokked by epoch 7,000, and mean final accuracy was 0.00653, below chance $1/113 = 0.00885$. Final unembedding difference alignment nevertheless exceeded addition alignment in every seed. Because this geometry arose in runs that never learned subtraction, the experiment cannot validate the diagnostic on that task and serves only as a negative scope check.

4.5 Additive attribution changes across tested conditions

The normalized analysis has established where task geometry is easiest to see. We now restore block magnitude and ask whether the same block contributes most of the task-aligned kernel mass. In the primary ReLU/no-LayerNorm condition, the two answers agree. From epochs 0 to 7,000, median normalized alignment changes by 0.5403 for the unembedding (95% bootstrap CI [0.3881, 0.6628]), compared with 0.0007 for embeddings, 0.0064 for attention, and 0.0017 for the MLP. On the additive scale, the unembedding again dominates: median $\Delta C = 0.1801$ ([0.0656, 0.3405]), versus 0.0120 ([0.0081, 0.0136]) for the three upstream blocks combined. It exceeds every upstream block in all 20 seeds (Holm-adjusted $p = 1.14 \times 10^{-5}$) and holds a median 90.8% of endpoint target mass.

That agreement need not survive a change of condition and landmark. **Prospectively locked confirmation:** we evaluate a four-block replication with GELU [Hendrycks and Gimpel, 2016] and pre-LayerNorm [Xiong et al., 2020], keeping $p = 113$, $\rho = 0.3$, width, optimizer, and probe construction fixed. The architecture and epoch-2,250 landmark were locked from the preceding sweep before evaluating the 20 new seeds. It again yields a 10/10 split; all seeds later grok by epoch 3,200 and finish above 0.99 accuracy.

The normalized story replicates decisively. The unembedding accounts for 98.65% of summed positive alignment change, completely separates the landmark groups (MW $p = 0.00018$, $r_{\text{rb}} = 1.00$),

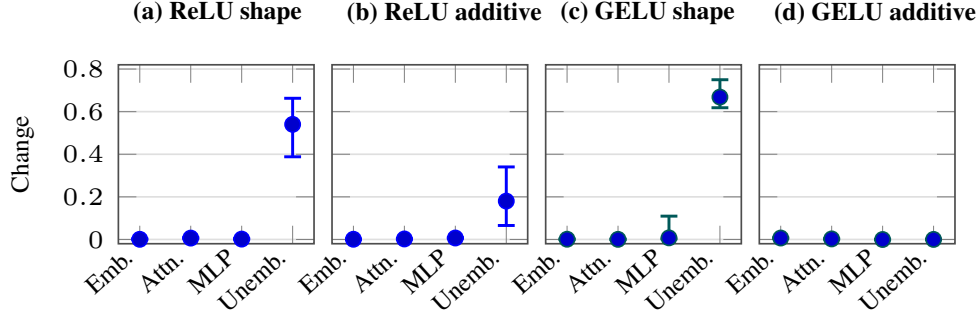


Figure 2: **Normalized readout dominance replicates, while additive localization differs across the tested condition–landmark pairs.** Each condition uses matched checkpoints: epoch 0 to 7,000 for ReLU/no-LayerNorm and epoch 0 to the locked 2,250 landmark for GELU+pre-LayerNorm. Points and error bars show medians with percentile-bootstrap 95% confidence intervals. All four panels use the same raw y-axis range $[-0.02, 0.82]$; the differing horizons preclude attribution of the reversal to architecture alone.

and tracks landmark accuracy ($\rho = 0.940$, $p = 7.27 \times 10^{-10}$). In every seed, addition alignment also exceeds both the per-seed permutation-null 95th percentile and the difference-coordinate control; the alternative metrics agree, with unembedding R^{sum} falling by 0.798 and E^{stat} by 0.527.

Additive attribution, however, now tells a different story. Under the same epoch-0-to-2,250 contrast, median $\Delta C^{(\ell)}$ is 0.006741 for embeddings (95% bootstrap CI $[0.004725, 0.008113]$), 0.002575 for attention ($[0.002045, 0.003102]$), 0.000237 for the unembedding ($[0.000153, 0.000477]$), and -0.000182 for the MLP ($[-0.000222, -0.000150]$). The ordering embedding > attention > unembedding > MLP holds in all 20 seeds, and all six paired contrasts survive Holm correction ($p_{\text{Holm}} = 1.14 \times 10^{-5}$; absolute paired rank-biserial correlation = 1).

The replication therefore separates visibility from additive ownership. The unembedding robustly dominates normalized representation-shape change, yet the GELU+pre-LayerNorm landmark assigns median endpoint shares of 66.91% to embeddings, 28.09% to attention, 2.85% to the unembedding, and 0.59% to the MLP (Figure 2). Because the horizons differ, this contrast rules out architecture-invariant attribution under the tested protocols without isolating architecture from training stage.

4.6 The representation emerges at the MLP output

Once visibility and additive ownership are separated, the original mechanistic question reemerges: where does the representation first arise in the forward pass? Applying the same centered sum-orbit metric to activation Grams at the epoch-7,000 primary checkpoints shows a distinct boundary. Addition alignment is minimal at the attention output and post-attention residual (mean -0.0082 at both stages), then increases to 0.5241 at the MLP output (95% bootstrap CI $[0.4181, 0.6241]$). It remains elevated in the final residual (0.4799, $[0.3682, 0.5843]$) and logits (0.5624, $[0.4633, 0.6590]$), while the difference-coordinate control stays near zero (Figure 3a). Thus, the representation emerges between the post-attention residual and MLP output, although its geometry is most clearly shown by the unembedding parameter-gradient kernel.

4.7 The eNTK-guided sum-orbit subspace is necessary for the evaluated solutions

Emergence by itself does not prove function, so we next perturb the rank-5 eNTK-guided centered sum-orbit subspace in the ten grokked primary models. Removal addresses the necessity question. Starting from an untouched mean accuracy of 0.9987, removing the guided subspace reduces accuracy to 0.1020 at the MLP output and 0.0981 at the final residual. Random rank-matched removal leaves accuracy essentially unchanged, while random rank-and-norm-matched removal reduces it to about 0.93. Guided removal impairs accuracy, loss, and correct-logit margin more than both controls at both stages (all locally Holm-adjusted $p = 0.00391$). With this immediate intervention, the selected geometry is thus necessary for the current computation in these ten evaluated grokked primary models; this does not imply necessity for grokking in general or exclusive mechanistic ownership.

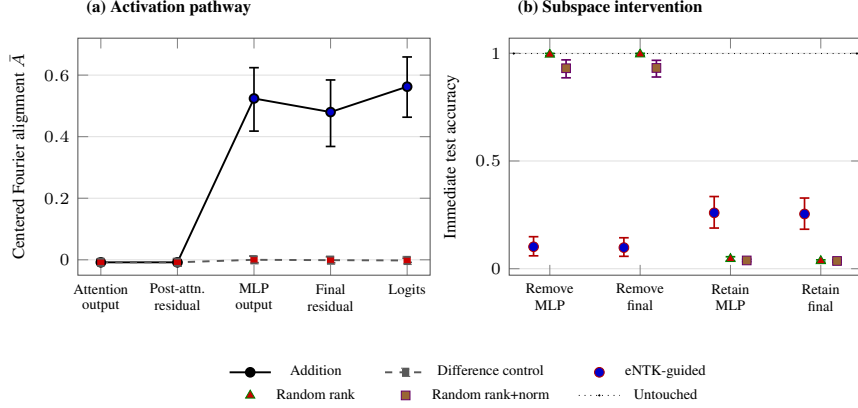


Figure 3: **Pathway and intervention results.** (a) Addition geometry appears sharply at the MLP output; the difference-coordinate control remains near zero. (b) Guided rank-5 removal is more damaging than random removal, while guided retention preserves more immediate accuracy than random retention. Error bars are seed-bootstrap 95% intervals.

Retention addresses the complementary sufficiency question, providing a weaker response. Retaining only the guided subspace maintains mean accuracy of 0.2594 at the MLP output and 0.2541 at the final residual, compared to about 0.04 for retained random subspaces. Guided retention surpasses either control at both stages (locally Holm-adjusted $p = 0.00391$), yet remains well below the untouched model, and its advantages in loss and margin are not consistently significant after adjustment. For these ten evaluated models under immediate intervention, the subspace is thus partially, not completely, sufficient for performance (Figure 3b). Effects at the MLP output and final residual are descriptively similar; we do not assert formal stage equivalence.

5 Why the Readout Reveals the Representation

The empirical sequence now indicates a consistent division of roles: the task representation emerges at the MLP output, continues to be detectable downstream, and is most clearly revealed by the readout Jacobian. The remaining question is why a parameter block that does not generate the representation can reveal it so distinctly. For the readout, the explanation comes directly from its Jacobian.

5.1 The readout kernel is a representation Gram matrix

Lin [2025] derives the per-class unembedding eNTK as a class-diagonal residual Gram matrix. To connect that result to our scalarized convention, write the unembedding as the linear map $W_U \in \mathbb{R}^{d_{\text{model}} \times p}$ applied to the final residual stream at the = token position. If $h_T(x) \in \mathbb{R}^{d_{\text{model}}}$ denotes that residual stream, then $f_c(x; \theta) = W_U[:, c]^\top h_T(x)$ and hence

$$\nabla_{W_U} f_c(x; \theta) = h_T(x) e_c^\top, \quad (3)$$

where e_c is the c -th standard basis vector. The scalarized unembedding-block eNTK satisfies

$$K_t^{(\text{unembed})}(x, x') = p \cdot \langle h_T(x), h_T(x') \rangle. \quad (4)$$

The class- c Frobenius product of these gradients is $\langle h_T(x), h_T(x') \rangle$, so summing over $c = 1, \dots, p$ proves Eq. 4. If $G_t = A_T A_T^\top$ is the residual Gram matrix, then $HK_U H = p H G_t H$ and $\text{Tr}(H K_U H) = p \text{Tr}(H G_t H)$, so the factor cancels exactly under trace-normalization. No assumption about the loss, optimizer, or training dynamics is involved: normalized unembedding geometry is normalized residual geometry. A seed-0 checkpoint comparison confirms agreement with the projected implementation to relative Frobenius error 2.88×10^{-6} . The readout eNTK is therefore a direct view of the representation presented to the decoder.

5.2 A Fourier representation makes the readout kernel align

The identity explains why the readout sees the residual representation, but not why that representation aligns with addition. The mechanistic circuit supplies the missing link. At post-grok checkpoints, the residual stream at the $=$ position is well approximated by a small number of Fourier features [Nanda et al., 2023]:

$$h_T(a, b) \approx \sum_{\omega \in \mathcal{K}} u_\omega \cos\left(\frac{2\pi\omega(a+b)}{p}\right) + v_\omega \sin\left(\frac{2\pi\omega(a+b)}{p}\right), \quad (5)$$

where $\mathcal{K}$ is a small set of key frequencies, typically 2–5 per model. If distinct-frequency directions are orthogonal and each sine/cosine pair is orthogonal with equal norm α_ω , the residual Gram reduces to $\sum_{\omega \in \mathcal{K}} \alpha_\omega \cos(2\pi\omega[s(x) - s(x')]/p)$ for $s(a, b) = a + b$, and is therefore addition-stationary. Norm imbalance, within-frequency nonorthogonality, and cross-frequency inner products produce the corresponding nonstationary remainder, yielding approximate stationarity when they are small. Together, Eqs. 3–5 explain the observed chain: the MLP produces sum-dependent Fourier features, the residual stream carries them forward, and the readout Jacobian converts their inner products directly into its kernel. The saturated six-orbit projector detects this dependence but does not identify the frequencies in $\mathcal{K}$.

Once the residual stream becomes organized by $a + b$, its Gram matrix is the natural place to observe sum structure. This explains both the strength of the unembedding alignment and its association with accuracy. The alignment measures organization of the final representation around the task; it does not relocate the origin of those features to the readout.

5.3 Why upstream gradient kernels can look different

The final piece of the puzzle is why the MLP parameter-gradient kernel need not display the feature that the MLP output contains. Upstream blocks do not enjoy the readout’s direct Gram identity. For a linear layer $y = Wx$ with input x_ℓ and $\delta_c = \partial f_c / \partial y$, entrywise differentiation gives $\partial f_c / \partial W_{ij} = \delta_{c,i} x_{\ell,j}$, hence $\nabla_W f_c = \delta_c x_\ell^\top$. Therefore

$$K^{(W)}(x, x') = \langle x_\ell(x), x_\ell(x') \rangle \sum_{c=1}^p \langle \delta_c(x), \delta_c(x') \rangle,$$

because the Frobenius product of the two outer products factors. A bias contributes only the second factor. An upstream gradient kernel therefore mixes input-side activation geometry with output-side backpropagated geometry. The MLP can construct a Fourier feature without its own parameter-gradient kernel becoming a clean Gram matrix of that feature. This factorization reconciles the weak MLP-kernel alignment with the sharp activation transition at the MLP output, while leaving neuron- and operation-level localization for future work.

6 Discussion

The apparent discrepancy between the MLP circuit and the unembedding eNTK shows that “where is the algorithm?” groups four questions together (Table 2). These responses provide a single explanation rather than alternative claims of ownership.

Table 2: **Four distinct localization claims supported by the evidence.**

Question	Answer supported here
Where does the representation emerge?	At the MLP output in the measured forward pathway.
Where is it easiest to observe in normalized eNTK geometry?	At the unembedding, whose kernel is the final-residual Gram.
Which block owns most additive target mass?	It depends on the tested condition and landmark.
Is the selected representation functionally important?	Yes: it is necessary and partially sufficient under the immediate intervention in ten evaluated grokked primary models.

Layerwise eNTK studies should thus specify the localization claim that their normalization, activation measurements, and interventions substantiate.

Scope of the evidence. The primary landmark is exploratory and dependent on budget, with the dose-response being contemporaneous. Additive analysis includes two conditions but excludes the modulus sweep; pathway Grams are endpoint-only; and the rank-5 intervention addresses ten grokked primary seeds. The locked subtraction control did not generalize and therefore functions solely as a negative scope check.

7 Limitations

Complete four-block evidence includes the primary model, one locked GELU+pre-LayerNorm confirmation, and six modulus-specific runs with varying budgets. Alternative symmetry metrics are mixed at $p = 113$ and unfavorable at $p = 127$, while the two condition–landmark pairs provide opposite additive orderings. Neither symmetry dominance nor block attribution is consistent across all cases.

The activation-subspace intervention is immediate, covers ten already-grokked primary models, and shows only partial sufficiency; no formal equivalence test compares the MLP-output and final-residual stages. The subtraction control produced no grokked seeds and supplies no positive task-transfer evidence. Although the random-control seed rule and draw-level outcomes are recovered, the original sampled basis tensors and software versions are unavailable, preventing a claim of bitwise reproduction. Thus the results localize a functionally important sum-dependent representation without identifying its frequencies, supplying a neuron-level circuit, or establishing task-general validity.

Broader impacts. This work studies small transformers on synthetic arithmetic and introduces no human data, deployed system, or high-capability model. Its potential benefit is methodological: distinguishing representation diagnostics from causal or additive attribution may encourage more careful mechanistic claims. The main foreseeable risk is scientific overinterpretation—for example, treating a strongly aligned readout Jacobian as proof that the readout implements the computation. We mitigate this risk by crediting the inherited readout identity, reporting condition-dependent attribution and the negative subtraction scope check, and restricting conclusions to the tested protocols and task.

8 Conclusion

The apparent conflict between the MLP circuit and the readout eNTK resolves into a more precise account of localization. The MLP output is where sum-orbit geometry emerges in the measured forward pathway; the readout Jacobian reveals it most clearly because its normalized kernel is the final-residual Gram. In ten grokked primary models, immediate removal of the eNTK-guided subspace nearly destroys accuracy, while retaining only it preserves a meaningful but incomplete part of performance. This supports necessity for those evaluated solutions and partial immediate sufficiency, not unique mechanistic ownership. Additive attribution supplies no single replacement location: it is readout-dominant at the primary endpoint and upstream-dominant at the locked GELU+pre-LayerNorm landmark. Thus visibility, additive contribution, pathway emergence, and intervention-based functional relevance remain distinct claims.

References

- S. Arora et al. On exact computation with an infinitely wide neural net. In *NeurIPS*, 2019.
- A. Atanasov, B. Bordelon, and C. Pehlevan. Neural networks as kernel learners: The silent alignment effect. *arXiv preprint arXiv:2111.00034*, 2021.
- J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- A. Canatar, B. Bordelon, and C. Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature Communications*, 12(1):2914, 2021.
- L. Chizat, E. Oyallon, and F. Bach. On lazy training in differentiable programming. In *NeurIPS*, 2019.
- M. Geiger, S. Spigler, A. Jacot, and M. Wyart. Disentangling feature and lazy training in deep neural networks. *JSTAT*, 2020.
- A. Gromov. Grokking modular arithmetic. *arXiv preprint arXiv:2301.02679*, 2023.
- D. Hendrycks and K. Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.
- A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *NeurIPS*, 2018.
- T. X. Khanh, T. Q. Hoa, L. D. Trung, and P. T. Duc. The norm-separation delay law of grokking: A first-principles theory of delayed generalization. *arXiv preprint arXiv:2603.13331*, 2026.
- T. Kumar, B. Bordelon, S. J. Gershman, and C. Pehlevan. Grokking as the transition from lazy to rich training dynamics. In *ICLR*, 2024.

- J. Lee et al. Wide neural networks of any depth evolve as linear models under gradient descent. In *NeurIPS*, 2019.
- J. Lin. Feature identification via the empirical NTK. *arXiv preprint arXiv:2510.00468*, 2025.
- I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.
- M. A. Mohamadi, W. Bae, and D. J. Sutherland. A fast, well-founded approximation to the empirical neural tangent kernel. In *ICML*, 2023.
- M. A. Mohamadi, Z. Li, L. Wu, and D. J. Sutherland. Why do you grok? a theoretical analysis of grokking modular addition. In *ICML*, 2024.
- V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *ICML*, pages 807–814, 2010.
- N. Nanda, L. Chan, T. Lieberum, J. Smith, and J. Steinhardt. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217*, 2023.
- R. Novak, J. Sohl-Dickstein, and S. S. Schoenholz. Fast finite width neural tangent kernel. In *ICML*, 2022.
- A. Power, Y. Burda, H. Edwards, I. Babuschkin, and V. Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.
- N. Rahaman et al. On the spectral bias of neural networks. In *ICML*, 2019.
- H. Shan and B. Bordelon. A theory of neural tangent kernel alignment and its influence on training. *arXiv preprint arXiv:2105.14301*, 2022.
- M. Tancik et al. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017.
- R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T.-Y. Liu. On layer normalization in the transformer architecture. In *ICML*, pages 10524–10533, 2020.
- G. Yang. Tensor programs II: Neural tangent kernel for any architecture. *arXiv preprint arXiv:2006.14548*, 2020.

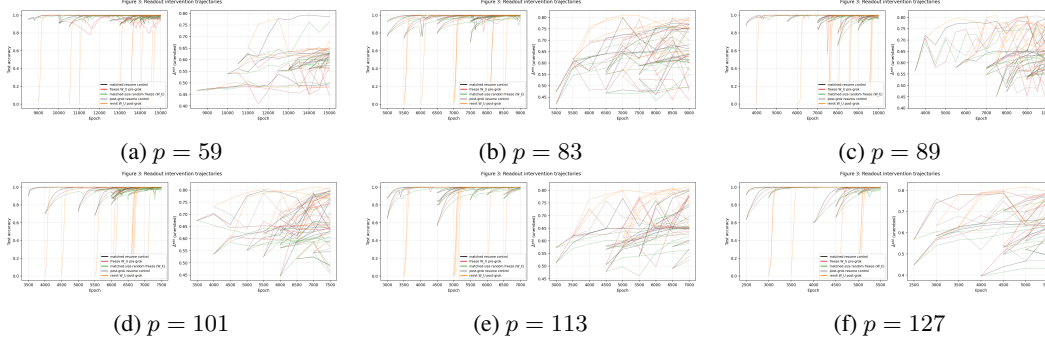


Figure 4: **Readout intervention trajectories across all completed moduli.** The panels compare matched resumes, readout freezing, embedding freezing, and post-grok readout reinitialization. The main quantitative intervention claims rely on the six-seed $p = 113$ subset; the other moduli serve as qualitative scope checks.

Table 3: **Final experiment status.** “Completed” indicates that the result is included in the final archive. Limitations of the causal-basis implementation record are explicitly stated below.

Item	Status
Modulus, rank, and projection checks	Completed; the modulus landmarks are exploratory and selection-conditional
Primary and GELU+pre-LN additive attribution	Completed at their respective locked analysis horizons
Training fractions and recipes	Grok counts, times, and exact-readout changes completed; four-block comparison not run
Endpoint activation pathway	Completed over all 20 primary seeds; the centered equals-token embedding Gram has zero trace
Rank-5 activation-subspace interventions	Completed on the ten grokked primary seeds; all 1,640 draw-level rows and deterministic random-basis seeds are recovered, but original basis tensors are absent
Task and null controls	Difference-coordinate and permutation controls completed; subtraction completed but produced no grokked seeds
Time-to-event analysis	Completed as an auxiliary calculation under the same recorded-checkpoint event rule; not used for causal claims
Prospective warning rule	Not run; retrospective trajectories are not presented as prospective prediction
Compute accounting	Completed with observed aggregate traces and reconstructed per-stage timing, memory, and project GPU-hour estimates

A Readout Intervention Trajectories Across Moduli

B Additional Scope Results and Final Experiment Status

The tables below present auxiliary sweeps that provide insight into grokking frequency and the precise residual/unembedding readout, followed by consolidated summaries of the final additive, activation, task-control, and intervention experiments. Each architecture, training-fraction, and recipe row employs 20 seeds unless otherwise indicated. The training-fraction and recipe sweeps do not estimate attention, embedding, or MLP parameter-block eNTKs and thus do not assess cross-setting blockwise dominance.

Compute accounting. All implementations use one GPU per process and have no distributed or multi-GPU path. The original experiments used NVIDIA RTX 6000 Pro and H100 GPUs, while one confirmed reproduction used an A100-SXM4-80GB. The exact original run-to-device mapping was not retained. Table 4 combines two observed aggregate traces with reconstructed per-stage timing, memory, and total-compute estimates. Ranges are plausible engineering ranges rather than confidence intervals or direct memory telemetry. They are based on the approximately 227,000-parameter model, 3,830 training examples, 8,939 test examples, 678 eNTK probe examples, and 16 Rademacher projections.

Seed and data provenance. The ReLU/no-LayerNorm addition runs use model seeds 0–19, the subtraction control uses 100–119, and the GELU+pre-LayerNorm confirmation uses 100–119. Initialization seeds Python’s random, the legacy NumPy global generator, the PyTorch CPU generator,

Table 4: **Observed and reconstructed compute information.** Values marked “estimate” report a best estimate followed by a plausible range.

Field	Reconstructed value
Worker configuration	One GPU per process; no distributed or multi-GPU execution
Hardware	Original RTX 6000 Pro and H100; confirmed reproduction A100-SXM4-80GB
Observed GELU+pre-LN trace	20 combined seeds in 28.0 min = 0.467 GPU-hours; GPU model unrecorded
Observed A100 trace	20 combined $p = 101$ seeds in 22.7 min = 0.378 A100 GPU-hours
A100 training, 5,000 epochs (estimate)	30 s per seed; plausible range 20–40 s
A100 training, 7,000 epochs (estimate)	42 s per seed; plausible range 30–55 s
eNTK checkpoint (estimate)	A100: 1.8 s (1.3–2.5); H100: 1.2 s (0.8–1.8); RTX 6000 Pro: 2.5 s (1.8–3.5)
GELU+pre-LN per seed (estimate)	30 s training + 38 s for 21 eNTKs = 68 s; plausible range 55–85 s
ReLU/no-LN per seed (estimate)	42 s training + 4 s for two eNTKs = 46 s; plausible range 35–60 s
Peak allocated memory (estimate)	2.0 GB; plausible range 1.5–3.0 GB
Peak reserved memory (estimate)	3.0 GB; plausible range 2.0–5.0 GB
Primary addition and subtraction	Approximately 0.5 GPU-hours
Activation and causal interventions (estimate)	0.075 GPU-hours; plausible range 0.05–0.10
Archived and reproduction workload (estimate)	1.4 GPU-hours; plausible range 1.1–1.9
Full project compute (estimate)	2.5 GPU-hours including exploratory, failed, and repeated runs; plausible range 2.0–4.0

and all CUDA generators. The dataset seed is 0: all 113^2 pairs are enumerated in lexicographic loop order, the equals token 113 is appended, and a CPU `torch.randperm(12769)` seeded with 0 assigns the first 3,830 indices to training and the remaining 8,939 to testing. Addition and subtraction use identical indices and change only the labels.

The projected eNTK employs Rademacher seed 10000: a CPU generator produces a 16×113 binary array, which is mapped to $\{-1, +1\}$ and reused across model seeds and checkpoints. The GELU landmark permutation test uses seed 72011 for 50 sequential permutations of the 678 probe samples. Percentile intervals involve 10,000 resamples from `numpy.random.default_rng`. The base seed is 91003 for endpoint summaries, 92003 for activation summaries, and 96003 for causal summaries, with deterministic offsets defined by the documented block, quantity, stage, metric, or group order. The primary and subtraction checkpoint manifests include epochs 0 and 7,000 and 140 evaluation records per completed seed; the GELU condition measures eNTKs every 250 epochs through epoch 5,000, with its landmark fixed at epoch 2,250.

Software and artifact provenance. The original Python, PyTorch, CUDA, cuDNN, driver, NumPy, SciPy, pandas, and Jupyter versions were not recorded; the released requirements provide lower bounds rather than a bitwise environment lock. A rendered GELU figure identifies Matplotlib 3.10.0, but does not establish the rest of the original environment. Consequently, numerical reproduction is supported while version-dependent bitwise identity is not claimed. The extracted artifact has no Git history from which to recover a submission commit. File-level checksums are supplied instead; in particular, the 1,640-row random-control table has SHA-256 0a3aa964ade2525bc55835a87d05171a07251fa1762295566f5dfe68a6042430.

Modulus and seed expansion. Broader modulus coverage should come with more seeds per modulus, not just more isolated rows. Table 5 keeps the measured $p = 59$, $p = 83$, $p = 89$, $p = 101$, $p = 113$, and $p = 127$ results visible while adding a descriptive pooled row over the completed sweep. In that row, seeds and grok counts are summed across moduli, median t_{grok} is taken over all grokked seeds, and within-block positive-change ratios are weighted by the number of grokked seeds in each modulus row. Because the horizons were chosen separately to obtain informative splits, we do not pool the six correlations into a single inferential statistic. A full mixed-effects analysis remains a separate follow-up rather than a completed result.

Training-fraction expansion. Training fraction changes both the occurrence and timing of the transition. No seed groks at $\rho = 0.20$ by epoch 7,000, the baseline gives 10/20 at $\rho = 0.30$, and all 20 seeds grok for $\rho \in \{0.40, 0.50, 0.70\}$. Median grok time falls from 1,225 epochs at $\rho = 0.40$ to 600 at $\rho = 0.50$ and 250 at $\rho = 0.70$. Endpoint accuracy is constant at 1.0 for the 0.40 and 0.50 rows, making their Spearman correlations undefined. Retrospective lead times and four-block ratios were not computed for the auxiliary rows.

Table 5: **Completed modulus and seed summary.** The pooled row represents a descriptive aggregate of completed modulus rows, rather than a new modulus or a mixed-effects model. Per-modulus landmark tests are exploratory, as their horizons were chosen to produce informative splits.

p	seeds (plan/actual)	grokked	med. t_{grok}	unemb. ratio	MLP ratio	Spearman $\rho(p)$
59	20/20	10	13425	69.0%	8.3%	0.868 (6.94×10^{-7})
83	20/20	10	6700	72.0%	0.2%	0.902 (5.61×10^{-8})
89	20/20	12	8100	68.5%	8.6%	0.926 (4.94×10^{-9})
101	20/20	12	6150	72.4%	0.5%	0.802 (2.14×10^{-5})
113	20/20	10	4900	95.8%	0.0%	0.928 (3.76×10^{-9})
127	20/20	9	4550	99.4%	0.0%	0.892 (1.31×10^{-7})
pooled across p	120/120	63	6400	78.6%	3.1%	not pooled

Table 6: **Completed training-fraction sweep.** ΔA_U represents the exact residual/unembedding-Gram alignment change, rather than a four-block ratio. All rows employ $p = 113$ with the baseline architecture and optimizer.

train frac.	seeds	grokked	med. t_{grok}	med. ΔA_U	$\rho(p)$
0.20	20/20	0	—	0.003	0.590 (0.00618)
0.30	20/20	10	4900	—	0.928 (3.76×10^{-9})
0.40	20/20	20	1225	0.796	undefined (ceiling)
0.50	20/20	20	600	0.804	undefined (ceiling)
0.70	20/20	20	250	0.767	0.026 (0.913)

Training-recipe expansion. Recipe fragility is pronounced. Extending the baseline recipe to 15,000 epochs yields 20/20 grokked seeds, while reducing weight decay to 0.1 or 0.3 yields no events by epoch 9,000. Increasing weight decay to 3.0 instead gives 20/20 events with median $t_{\text{grok}} = 2,275$. Lowering the learning rate gives 1/20, cosine scheduling gives 2/20, and Adam without weight decay gives 0/20. The exact readout change remains strongly correlated with partial final accuracy for the lower-learning-rate and cosine rows, but these outcomes do not establish four-block localization stability.

Additive target-mass attribution. The primary ReLU/no-LayerNorm and GELU+pre-LayerNorm experiments both use full-trace-normalized blocks $K_t^{*(\ell)} = HK_t^{(\ell)}H/\text{Tr}(HK_tH)$ and report $C_t^{(\ell)} = \text{Tr}(P_{\text{add}}^0 K_t^{*(\ell)})$. Table 8 shows that the additive ordering differs between the two tested architecture-landmark conditions. Since the horizons also vary, this comparison does not isolate architecture as the reason for the reversal.

Auxiliary censored time-to-grok analysis. The final archive includes a Kaplan–Meier and Weibull accelerated-failure-time comparison of addition and subtraction based on the paper’s criterion: the first evaluation in two consecutive recorded evaluations above 0.95. Addition has 10/20 events by epoch 7,000 and subtraction has 0/20; the Weibull time-ratio estimate equals 5.39 with a wide 95% interval $[0.46, 63.61]$ and $p = 0.181$. Given complete censoring in the subtraction arm, this auxiliary comparison remains inconclusive and is not employed for the main causal claims.

Prospective warning-rule validation (not performed). No threshold was calibrated on pilot seeds or evaluated on held-out seeds. Accordingly, the paper does not report ROC-AUC, precision, false-alarm rates, or prospective lead times.

Endpoint activation-Gram pathway. Proposition 14 identifies $K_t^{(\text{unembed})}$ with the final residual-stream Gram matrix after normalization, but it does not by itself identify where that representation first forms. The final endpoint analysis covers all 20 primary seeds. The equals-token embedding output is constant across examples, so its centered Gram has zero trace and its normalized alignment

Table 7: **Completed training-recipe sweep.** ΔA_U represents the median exact residual/unembedding-Gram alignment change from epoch 0 to the final checkpoint. It is not a four-block positive-change ratio. The rows use $p = 113$, $\rho = 0.3$, and the baseline architecture.

recipe	optimizer	lr	wd	epochs	seeds	grokked	med. t_{grok}	med. ΔA_U	$\rho(p)$
baseline	AdamW	10^{-3}	1.0	7000	20/20	10	4900	—	0.928 (3.76×10^{-9})
longer budget	AdamW	10^{-3}	1.0	15000	20/20	20	7075	0.745	−0.104 (0.662)
lower wd	AdamW	10^{-3}	0.1	9000	20/20	0	—	0.007	0.346 (0.135)
medium wd	AdamW	10^{-3}	0.3	9000	20/20	0	—	0.035	0.568 (0.00900)
higher wd	AdamW	10^{-3}	3.0	9000	20/20	20	2275	0.812	undefined (ceiling)
lower lr	AdamW	3×10^{-4}	1.0	15000	20/20	1	13000	0.128	0.883 (2.59×10^{-7})
cosine schedule	AdamW	10^{-3}	1.0	9000	20/20	2	3850	0.247	0.879 (3.31×10^{-7})
Adam no wd	Adam	10^{-3}	0.0	9000	20/20	0	—	0.005	0.302 (0.195)

Table 8: **Matched-within-condition additive summaries.** Values are seed medians. Endpoint shares are percentages of $\sum_{\ell} C_T^{(\ell)}$; changes compare epoch 0 with the stated endpoint.

Condition	quantity	embedding	attention	MLP	unembedding
ReLU/no-LN, $T = 7000$	$\Delta C^{(\ell)}$	0.000956	0.002364	0.006884	0.180124
ReLU/no-LN, $T = 7000$	endpoint share	0.67%	1.26%	7.30%	90.8%
GELU+pre-LN, $T = 2250$	$\Delta C^{(\ell)}$	0.006741	0.002575	−0.000182	0.000237
GELU+pre-LN, $T = 2250$	endpoint share	66.9%	28.1%	0.59%	2.85%

is undefined. Alignment is negligible through attention and the post-attention residual, rises sharply at the MLP output, and remains high in the final residual and logits (Table 9). A separate retrospective analysis on the ten grokked seeds detected a very small earlier post-attention change, so the endpoint result locates the sharp geometric transition rather than proving a unique first computational operation.

Rank-5 activation-subspace interventions. Projection and ablation tests were completed on the ten grokked primary models. Mean untouched accuracy is 0.9987. Removing the eNTK-guided sum-orbit basis reduces mean accuracy to 0.1020 at the MLP output and 0.0981 at the final residual, whereas removing random rank-5 directions leaves 0.9944 and 0.9954 accuracy; norm-matched random removal leaves 0.9304 and 0.9317. Retaining only the guided basis preserves 0.2594 and 0.2541, while random retention preserves approximately 0.036–0.046. Paired tests use Holm correction over the two predeclared controls within each stage, intervention mode, and outcome. Under this immediate intervention, these results support necessity for the ten evaluated grokked primary solutions and partial, not full, sufficiency for their immediate performance; they do not establish necessity for grokking generally or unique mechanistic ownership. Appendix C.9 specifies the sample-space-to-activation-space construction and the recovered deterministic rule for all 20 random replicates.

Matched block perturbations (not performed). The completed interventions cover W_U and an approximate embedding control, not matched attention and MLP perturbations.

Completed and remaining task controls. The GELU+pre-LayerNorm confirmation incorporates a difference-coordinate task-mismatch control along with per-seed permutation nulls. A locked modular-subtraction control was also performed, but no seed was grokked within the analysis horizon; thus, it functions solely as a negative scope check and offers no positive evidence of task transfer. Harder group operations and random-label controls were not conducted. The specificity claim is limited to modular addition and the reported mismatch and permutation controls.

Statistical limitations. The final analysis presents seed-bootstrap intervals for the main endpoint, additive, activation, and intervention summaries, as well as paired and permutation tests where indicated. The primary and modulus horizons were still chosen to produce informative splits, and their landmark tests remain exploratory; only the GELU landmark was fixed before assessing the new seeds. The causal-basis multiplicity correction applies locally rather than across the entire study. The

Table 9: **Endpoint activation-Gram pathway at $p = 113$.** Entries are medians with seed-bootstrap 95% intervals over all 20 primary seeds. The difference-coordinate column is a task-mismatch control.

Stage	addition alignment	difference control
equals-token embedding output	undefined (zero centered trace)	undefined
attention output	$-0.008235 [-0.008470, -0.007977]$	-0.008103
post-attention residual	$-0.008235 [-0.008465, -0.007976]$	-0.008103
MLP output	$0.524118 [0.418088, 0.624149]$	-0.000014
final residual	$0.479919 [0.368189, 0.584254]$	-0.001173
logits	$0.562410 [0.463261, 0.658966]$	-0.002133

result archive includes exact guided values, all random-control draws, per-seed control means, and deterministic basis seeds; bitwise regeneration is restricted only by the lack of the original PyTorch version and sampled basis tensors.

C Complete Task-Specific eNTK Method

This section outlines the full eNTK pipeline applied in the addition and subtraction experiments. It distinguishes four elements that are often confused: the exact scalarized eNTK, its random-output-projection estimator, independently normalized block shapes, and the shared-denominator additive decomposition. All Jacobians are computed at a fixed checkpoint; neither the training-loss Hessian nor optimizer state influences the eNTK.

C.1 Forward map and parameter blocks

For an input pair $(a, b) \in \mathbb{Z}_p^2$, the token sequence is $x = (a, b, =)$, where the equals token has index p . Let $X_0(x) \in \mathbb{R}^{3 \times d}$ be the sum of token and positional embeddings. For attention head h ,

$$Q_h = \mathcal{N}_{\text{attn}}(X_0)W_Q^{(h)}, \quad K_h = \mathcal{N}_{\text{attn}}(X_0)W_K^{(h)}, \quad V_h = \mathcal{N}_{\text{attn}}(X_0)W_V^{(h)},$$

$$A_h = \text{softmax}\left(\frac{Q_h K_h^\top}{\sqrt{d_h}} + M_{\text{causal}}\right), \quad O = \sum_h A_h V_h W_O^{(h)}, \quad X_1 = X_0 + O.$$

Here M_{causal} is zero on allowed positions and $-\infty$ above the causal diagonal. The MLP and readout are

$$U = \phi(\mathcal{N}_{\text{mlp}}(X_1)W_{\text{in}} + b_{\text{in}}),$$

$$X_2 = X_1 + UW_{\text{out}} + b_{\text{out}},$$

$$h_T(x) = X_2(x)_{=,:}, \quad f(x; \theta) = h_T(x)^\top W_U \in \mathbb{R}^p.$$

For the primary and modulus experiments, $\mathcal{N}_{\text{attn}}$ and $\mathcal{N}_{\text{mlp}}$ are the identity and ϕ is ReLU. For the locked replication they are pre-LayerNorm operations and ϕ is GELU. Attention and the unembedding are bias-free; the MLP contains the displayed biases.

The parameter vector is divided into disjoint coordinates

$$\theta = (\theta^{(\text{embed})}, \theta^{(\text{attn})}, \theta^{(\text{mlp})}, \theta^{(\text{unembed})}).$$

For the ReLU/no-LayerNorm model, the blocks consist of

$$\theta^{(\text{embed})} = (W_E, W_{\text{pos}}),$$

$$\theta^{(\text{attn})} = (W_Q, W_K, W_V, W_O),$$

$$\theta^{(\text{mlp})} = (W_{\text{in}}, b_{\text{in}}, W_{\text{out}}, b_{\text{out}}),$$

$$\theta^{(\text{unembed})} = W_U.$$

In the pre-LayerNorm replication, the pre-attention LayerNorm gain and bias are attributed to the attention block, and the pre-MLP LayerNorm gain and bias to the MLP block. This maintains a disjoint partition and ensures the raw four-block sum is exact.

C.2 Exact scalarized and blockwise eNTKs

For block ℓ with d_ℓ scalar parameters, define its output Jacobian

$$J_t^{(\ell)}(x) = \frac{\partial f(x; \theta_t)}{\partial \theta^{(\ell)}} \in \mathbb{R}^{p \times d_\ell}.$$

The exact scalarized block eNTK is

$$K_t^{(\ell)}(x, x') = \left\langle J_t^{(\ell)}(x), J_t^{(\ell)}(x') \right\rangle_F = \sum_{c=1}^p \langle \nabla_{\theta^{(\ell)}} f_c(x), \nabla_{\theta^{(\ell)}} f_c(x') \rangle. \quad (6)$$

Proposition 2 (Exact block additivity). *For any disjoint Euclidean parameter partition, the scalarized eNTK satisfies*

$$K_t(x, x') = \sum_{\ell} K_t^{(\ell)}(x, x').$$

The same equality holds pointwise for the projected estimator below when the same output projections are used for every block.

Proof. For each output class c , the gradient in the direct-sum parameter space is the concatenation

$$\nabla_{\theta} f_c(x) = \bigoplus_{\ell} \nabla_{\theta^{(\ell)}} f_c(x).$$

The Euclidean inner product on a direct sum is the sum of the component inner products, so

$$\langle \nabla_{\theta} f_c(x), \nabla_{\theta} f_c(x') \rangle = \sum_{\ell} \langle \nabla_{\theta^{(\ell)}} f_c(x), \nabla_{\theta^{(\ell)}} f_c(x') \rangle.$$

Summing over c proves the exact identity. For a fixed projected scalar output, its full gradient is likewise the concatenation of its block gradients; applying the same argument before summing over projections proves projected additivity. There are no cross-block terms because distinct blocks occupy orthogonal coordinate subspaces of the parameter vector. $\square$

C.3 Random-output-projection estimator

At each eNTK checkpoint, draw independent vectors $z_1, \dots, z_R \in \{-1, +1\}^p$ with independent Rademacher entries. The same vectors are reused for every probe example and every parameter block at that checkpoint. Define

$$g_r(x; \theta) = \frac{1}{\sqrt{R}} z_r^\top f(x; \theta), \quad G_r^{(\ell)}(x) = \nabla_{\theta^{(\ell)}} g_r(x; \theta_t).$$

The estimator is

$$K_{t,R}^{\text{RP},(\ell)}(x, x') = \sum_{r=1}^R \left\langle G_r^{(\ell)}(x), G_r^{(\ell)}(x') \right\rangle. \quad (7)$$

In implementation, per-example gradients are vectorized over probe chunks of 64 examples, flattened within each parameter block, and accumulated as Gram matrices. The primary and modulus runs use $R = 16$ and eNTK checkpoints every 500 epochs; the projection-count check repeats selected checkpoints with $R \in \{32, 64\}$. The GELU+pre-LayerNorm replication uses $R = 16$, chunk size 64, and checkpoints every 250 epochs.

Proposition 3 (Unbiasedness and positive semidefiniteness). *For every block ℓ and pair x, x' ,*

$$\mathbb{E}_z \left[K_{t,R}^{\text{RP},(\ell)}(x, x') \right] = K_t^{(\ell)}(x, x').$$

For any realized projections, the probe-set matrix $K_{t,R}^{\text{RP},(\ell)}$ is symmetric positive semidefinite. Moreover,

$$K_{t,R}^{\text{RP}} = \sum_{\ell} K_{t,R}^{\text{RP},(\ell)}.$$

Proof. Suppress t and ℓ , and write $J(x) \in \mathbb{R}^{p \times d}$. By the chain rule,

$$G_r(x) = \frac{1}{\sqrt{R}} J(x)^\top z_r.$$

Therefore

$$K_R^{\text{RP}}(x, x') = \frac{1}{R} \sum_{r=1}^R z_r^\top J(x) J(x')^\top z_r.$$

Rademacher entries satisfy $\mathbb{E}[z_{r,i} z_{r,j}] = \mathbf{1}\{i = j\}$, hence $\mathbb{E}[z_r z_r^\top] = I_p$. Using $\mathbb{E}[z^\top B z] = \text{Tr}(B)$ for any square B ,

$$\begin{aligned} \mathbb{E}[K_R^{\text{RP}}(x, x')] &= \text{Tr}(J(x) J(x')^\top) \\ &= \text{Tr}(J(x')^\top J(x)) \\ &= \langle J(x), J(x') \rangle_{\text{F}} = K(x, x'). \end{aligned}$$

For positive semidefiniteness, concatenate the projected gradient features over r :

$$\Phi(x) = (G_1(x)^\top, \dots, G_R(x)^\top)^\top.$$

Then $K_R^{\text{RP}}(x, x') = \langle \Phi(x), \Phi(x') \rangle$, so every finite probe matrix is a Gram matrix. The final equality follows from Proposition 2, applied to each projected scalar g_r . $\square$

Proposition 4 (The projected unembedding estimator is exact). *For a bias-free linear unembedding and Rademacher output projections scaled as in Eq. 7,*

$$K_{t,R}^{\text{RP},(\text{unembed})}(x, x') = p \langle h_T(x), h_T(x') \rangle$$

for each realized set of projections and every $R \geq 1$; thus, this block exhibits no output-projection sampling variance.

Proof. For $g_r(x) = R^{-1/2} z_r^\top W_U^\top h_T(x)$,

$$\nabla_{W_U} g_r(x) = \frac{1}{\sqrt{R}} h_T(x) z_r^\top.$$

Consequently,

$$\begin{aligned} \langle \nabla_{W_U} g_r(x), \nabla_{W_U} g_r(x') \rangle_{\text{F}} &= \frac{\|z_r\|^2}{R} \langle h_T(x), h_T(x') \rangle \\ &= \frac{p}{R} \langle h_T(x), h_T(x') \rangle, \end{aligned}$$

because every Rademacher vector has exactly $\|z_r\|^2 = p$. Summing the identical contribution over $r = 1, \dots, R$ gives the result. Proposition 14 independently derives the same identity by summing the classwise exact Jacobians. $\square$

C.4 Centering, independent normalization, and additive normalization

Let $S = (x_1, \dots, x_n)$ be a probe and $H = I - n^{-1} \mathbf{1} \mathbf{1}^\top$. For a block with positive centered trace, define

$$d_t^{(\ell)} = \text{Tr}(H K_t^{(\ell)} H), \quad \widehat{K}_t^{(\ell)} = \frac{H K_t^{(\ell)} H}{d_t^{(\ell)}}.$$

Let $D_t = \text{Tr}(H K_t H) = \sum_\ell d_t^{(\ell)}$ and

$$K_t^{\star(\ell)} = \frac{H K_t^{(\ell)} H}{D_t}.$$

Proposition 5 (Relation between shape and additive decompositions). *When all displayed denominators are positive,*

$$\begin{aligned}\tilde{K}_t &= \frac{HK_t H}{D_t} = \sum_{\ell} K_t^{\star(\ell)} = \sum_{\ell} w_t^{(\ell)} \hat{K}_t^{(\ell)}, \\ w_t^{(\ell)} &= \frac{d_t^{(\ell)}}{D_t}, \quad \sum_{\ell} w_t^{(\ell)} = 1.\end{aligned}$$

For any orthogonal projector P ,

$$\begin{aligned}C_t^{(\ell)} &= \text{Tr}(PK_t^{\star(\ell)}) \geq 0, \\ \sum_{\ell} C_t^{(\ell)} &= \text{Tr}(P\tilde{K}_t).\end{aligned}$$

If the last total is positive, the shares $s_t^{(\ell)} = C_t^{(\ell)} / \sum_j C_t^{(j)}$ sum to one. Endpoint differences $\Delta C^{(\ell)}$ can have either sign.

Proof. By Proposition 2 and linearity of multiplication by H ,

$$HK_t H = \sum_{\ell} HK_t^{(\ell)} H.$$

Taking traces proves $D_t = \sum_{\ell} d_t^{(\ell)}$. Dividing the matrix equality by D_t yields the shared-denominator sum. Since

$$K_t^{\star(\ell)} = \frac{d_t^{(\ell)}}{D_t} \frac{HK_t^{(\ell)} H}{d_t^{(\ell)}},$$

the weighted independent-normalization identity follows, and the weights sum to one.

Each centered block kernel is positive semidefinite because $v^{\top} HK_t^{(\ell)} H v = (Hv)^{\top} K_t^{(\ell)} (Hv) \geq 0$. Thus $K_t^{\star(\ell)}$ is positive semidefinite. If P is an orthogonal projector, then

$$\text{Tr}(PK_t^{\star(\ell)}) = \text{Tr}(PK_t^{\star(\ell)} P)$$

is the trace of a positive semidefinite matrix and is nonnegative. Linearity of trace gives the target-mass sum. The share identity follows by division by the positive total. Finally, a difference of two nonnegative endpoint masses need not be nonnegative, so signed $\Delta C^{(\ell)}$ values are permitted. $\square$

This proposition clarifies the estimands explicitly. Comparing $\hat{K}_t^{(\ell)}$ concerns each block's shape after discarding its magnitude. Additive attribution instead employs $K_t^{\star(\ell)}$ or, equivalently, the trace weights $w_t^{(\ell)}$ combined with block shape. Independently normalized blocks do not sum to $\tilde{K}_t$ themselves unless their trace weights are equal and the corresponding scaling is reinstated.

C.5 Task coordinates, closed probes, and projectors

Both reported tasks are instances of a linear coordinate over the finite field $\mathbb{Z}_p$:

$$q_{\alpha, \beta}(a, b) = \alpha a + \beta b \pmod{p}.$$

Modular addition uses $q_+(a, b) = a + b$ and modular subtraction uses $q_-(a, b) = a - b$. For the odd prime moduli in the paper, define the complete coordinate orbit

$$\mathcal{O}_{\tau}^q = \{(a, b) \in \mathbb{Z}_p^2 : q(a, b) = \tau\}.$$

If $\beta \neq 0$, enumerate it without rejection by

$$(a, b) = (a, \beta^{-1}(\tau - \alpha a)), \quad a = 0, \dots, p-1.$$

Thus addition uses $b = \tau - a$, while subtraction uses $b = a - \tau$. The task probe is

$$S_q = \bigcup_{\tau \in \Omega} \mathcal{O}_{\tau}^q, \quad \Omega = \{0, 17, 30, 52, 87, 88\} \pmod{p},$$

with duplicate residues removed if necessary. In every reported modulus row the six residues are distinct, so $|S_q| = 6p$. The within-orbit transformation

$$T_\delta^q(a, b) = (a + \beta\delta, b - \alpha\delta)$$

preserves q . For addition this is $(a + \delta, b - \delta)$; for subtraction it is the same-sign translation $(a - \delta, b - \delta)$, equivalent after replacing δ by $-\delta$ to $(a + \delta, b + \delta)$.

For any coordinate q evaluated on any finite probe S , let $Z_q \in \{0, 1\}^{|S| \times r_q}$ be the incidence matrix of the distinct observed values of q : one column per observed residue and one nonzero per row. Its orbit-constant projector is

$$P_q = Z_q(Z_q^\top Z_q)^{-1} Z_q^\top, \quad P_q^0 = H P_q H. \quad (8)$$

On a union of r complete q -orbits, $Z_q^\top Z_q = pI_r$, so $P_q = p^{-1} Z_q Z_q^\top$, $\text{rank}(P_q) = r$, and $P_q^0 = P_q - n^{-1} \mathbf{1}\mathbf{1}^\top$ has rank $r - 1$.

Proposition 6 (Addition and subtraction task/control ranks). *On the addition probe S_{q_+} formed from r complete sum-orbits,*

$$\text{rank}(P_{q_+}) = r, \quad \text{rank}(P_{q_-}) = p.$$

On the subtraction probe S_{q_-} formed from r complete difference-orbits,

$$\text{rank}(P_{q_-}) = r, \quad \text{rank}(P_{q_+}) = p.$$

Therefore the task projector has rank 6 and centered rank 5, while the mismatched-coordinate control spans all p observed values on the same probe. Each alignment uses the actual rank of its own projector in the null baseline and ceiling.

Proof. The task-coordinate rank follows because the r incidence columns have disjoint, nonempty supports. Consider one complete addition orbit $a + b = \tau$. Along that orbit,

$$q_-(a, b) = a - b = a - (\tau - a) = 2a - \tau.$$

Because p is odd, multiplication by 2 is a bijection on $\mathbb{Z}_p$. As a ranges over $\mathbb{Z}_p$, $2a - \tau$ therefore takes every residue exactly once. Hence q_- takes all p values on even one complete addition orbit; its incidence matrix on S_{q_+} has p disjoint nonempty columns and rank p .

Conversely, on a complete subtraction orbit $a - b = \tau$, we have $b = a - \tau$ and

$$q_+(a, b) = a + b = 2a - \tau.$$

The same bijection shows that addition takes every residue exactly once per subtraction orbit, so the addition-control incidence matrix has rank p . Centering removes the constant incidence direction from the rank- r task projector, leaving rank $r - 1$. $\square$

The character, invariance, and stationarity constructions also transfer without a hidden task-specific step. For $q \in \{q_+, q_-\}$ define

$$\psi_{q,\omega}(x) = \exp(2\pi i \omega q(x)/p)$$

and call a kernel q -stationary when $K(x, x') = \kappa(q(x) - q(x'))$. On r complete q -orbits, the Vandermonde argument in Lemma 1 gives an r -dimensional restricted character span. The task-specific symmetry diagnostics are

$$R_t^q = \frac{1}{p} \sum_{\delta \in \mathbb{Z}_p} \frac{\|\tilde{K}_t - (P_\delta^q)^\top \tilde{K}_t P_\delta^q\|_F}{\|\tilde{K}_t\|_F},$$

$$E_t^{\text{stat},q} = \frac{\|\tilde{K}_t - \Pi_{\text{stat},q}(\tilde{K}_t)\|_F}{\|\tilde{K}_t\|_F},$$

where P_δ^q is induced by T_δ^q and $\Pi_{\text{stat},q}$ averages entries over classes with equal $q(x) - q(x')$. Proposition 11 proves orthogonality of this averaging projector verbatim after replacing s by q .

Table 10: **Complete eNTK execution record by task family.** “Exact readout” means that only the closed-form unembedding block pG_t was required; it must not be interpreted as a four-block eNTK analysis.

Experiment	target/probe	model	eNTK schedule	estimator and comparisons
Primary addition	q_+ / six q_+ -orbits	ReLU, no LN, $p = 113$, seeds 0–19	every 500 epochs to 7,000	Four blocks, $R = 16$; P_{q_+} target, P_{q_-} mismatch; exact pG_t check for readout
Addition modulus sweep	q_+ / six q_+ -orbits	same model; six listed moduli	every 500 epochs to each row’s horizon	Four blocks, $R = 16$; selected checks at $R = 32, 64$
Locked addition replication	q_+ / six q_+ -orbits	GELU, pre-LN, $p = 113$, seeds 100–119	every 250 epochs; locked landmark 2,250	Four blocks, $R = 16$; difference-coordinate control and 50 permutation nulls
Locked subtraction control (failed to generalize)	q_- / six q_- -orbits	ReLU, no LN, $p = 113$, seeds 100–119	blockwise endpoints at epochs 0 and 7,000	Four blocks, $R = 16$; P_{q_-} target and P_{q_+} mismatch; negative scope check only
Fraction/recipe additions	q_+ / six q_+ -orbits	stated auxiliary configurations	alignment cadence stated per configuration	Exact readout block pG_t only; no four-block attribution claim

Proposition 7 (Spectral identity for either reported task coordinate). *For $q \in \{q_+, q_-\}$ and a full-grid q -stationary kernel,*

$$K(x, x') = \kappa(q(x) - q(x')),$$

the characters $\psi_{q,\omega}$ are eigenfunctions with eigenvalues

$$\lambda_\omega = p \sum_{u \in \mathbb{Z}_p} \kappa(u) e^{-2\pi i \omega u / p}.$$

The full-grid kernel rank is at most p , its restriction to r complete task orbits has rank at most r , and it is invariant under independent within-orbit transformations in either argument.

Proof. For either q_+ or q_- , every residue $t \in \mathbb{Z}_p$ has exactly p preimages in $\mathbb{Z}_p^2$. Also $q(T_\delta^q x) = q(x)$. Hence independent within-orbit transformations preserve K .

For the finite-grid operator $(\mathcal{K}g)(x) = \sum_{x'} K(x, x')g(x')$, group examples by $t = q(x')$:

$$\begin{aligned} (\mathcal{K}\psi_{q,\omega})(x) &= p \sum_{t \in \mathbb{Z}_p} \kappa(q(x) - t) e^{2\pi i \omega t / p} \\ &= p e^{2\pi i \omega q(x) / p} \sum_{u \in \mathbb{Z}_p} \kappa(u) e^{-2\pi i \omega u / p} \\ &= \lambda_\omega \psi_{q,\omega}(x). \end{aligned}$$

Finally let Z be the full-grid incidence matrix of the p values of q and set $C_{st} = \kappa(s - t)$. Then $K = ZCZ^\top$, so $\text{rank}(K) \leq p$. Restricting to r complete task orbits keeps only r incidence columns and gives rank at most r . $\square$

For a projector P_q of rank r_q and a rank- m leading eigenspace projector Q_m , the reported affine score is

$$\bar{A}_m(q; K) = \frac{m^{-1} \text{Tr}(P_q Q_m) - r_q / n}{\min(m, r_q) / m - r_q / n}.$$

For centered kernels, Proposition 13 provides the corresponding centered ceiling and null baseline. The addition experiments employ P_{q_+} as the target and P_{q_-} as the mismatch control. The subtraction experiment swaps these roles. The GELU permutation null samples a sample-index permutation matrix Π , replaces P_{q_+} by $\Pi^\top P_{q_+} \Pi$, and recalculates the score; conjugation maintains projector rank and eigenvalues while disrupting its connection to the task labels. Fifty fixed-seed permutations are applied per reported checkpoint.

C.6 Task-by-task execution record

For every four-block row, the numerical order is: (i) evaluate logits on the ordered task probe; (ii) generate the checkpoint’s fixed Rademacher matrix; (iii) differentiate each projected scalar output for every example; (iv) concatenate gradient coordinates within each block; (v) form one block Gram per projection and sum over projections; (vi) sum raw blocks for the full eNTK; (vii) center; and (viii) either normalize each block by its own centered trace for shape metrics or by the full centered trace for additive target mass. Symmetrization before eigendecomposition uses $(K + K^\top)/2$ to remove numerical antisymmetry. Eigenvectors are ordered by decreasing eigenvalue, and SVD/eigenvalue directions below 10^{-10} of the largest retained singular/eigenvalue are treated as numerical zero when a rank decision is required.

C.7 Linear-layer Jacobian factorization used in the interpretation

Proposition 8 (Jacobian kernel of an upstream linear layer). *Let a layer use $y(x) = Wx(x)$, and let $\delta_c(x) = \partial f_c(x)/\partial y(x)$. Then*

$$\nabla_W f_c(x) = \delta_c(x)x(x)^\top$$

and the weight-block scalarized eNTK is

$$K^{(W)}(x, x') = \langle x(x), x(x') \rangle \sum_{c=1}^p \langle \delta_c(x), \delta_c(x') \rangle.$$

If the layer has a bias, its bias-block contribution is

$$K^{(b)}(x, x') = \sum_{c=1}^p \langle \delta_c(x), \delta_c(x') \rangle.$$

Proof. For entries W_{ij} , the chain rule gives

$$\frac{\partial f_c}{\partial W_{ij}} = \sum_k \frac{\partial f_c}{\partial y_k} \frac{\partial y_k}{\partial W_{ij}} = \delta_{c,i} x_j,$$

because $\partial y_k / \partial W_{ij} = \mathbf{1}\{k = i\}x_j$. Hence $\nabla_W f_c = \delta_c x^\top$. The Frobenius product of two outer products factors:

$$\langle \delta_c(x)x(x)^\top, \delta_c(x')x(x')^\top \rangle_F = \langle \delta_c(x), \delta_c(x') \rangle \langle x(x), x(x') \rangle.$$

Summing over classes proves the weight identity. Since $\partial y / \partial b = I$, $\nabla_b f_c = \delta_c$, giving the bias identity. This establishes the main-text statement that an upstream parameter-gradient kernel combines input-side activation geometry with output-side backpropagated geometry rather than equaling the Gram matrix of the feature produced by the layer. $\square$

C.8 Activation-Gram eigenspace identity

Proposition 9 (Centered activation Grams and singular vectors). *Let $A \in \mathbb{R}^{n \times d}$ contain one activation vector per probe example and let $A_c = HA$. If $A_c = U\Sigma V^\top$ is an SVD, then*

$$HAA^\top H = A_c A_c^\top = U\Sigma^2 U^\top.$$

Thus the leading sample-space eigenspace used for activation-Gram alignment is exactly the leading left-singular subspace of the centered activation matrix. If $\|A_c\|_F > 0$, trace-normalization divides the eigenvalues by $\|A_c\|_F^2$ and leaves every eigenspace unchanged.

Proof. Since H is symmetric and idempotent,

$$HAA^\top H = (HA)(HA)^\top = A_c A_c^\top.$$

Substituting the SVD gives

$$A_c A_c^\top = U\Sigma V^\top V\Sigma^\top U^\top = U\Sigma^2 U^\top.$$

Also $\text{Tr}(A_c A_c^\top) = \|A_c\|_F^2$. Division by this positive scalar changes eigenvalues but not eigenvectors. $\square$

The equals-position embedding activation remains the same for all probe examples since the final input token is always = and the positional embedding is constant. Consequently, its centered activation matrix is zero, making its normalized activation Gram and alignment undefined instead of zero. Every other activation-pathway stage employs Proposition 9 along with the same task/control projectors as their respective tasks.

C.9 eNTK-guided activation-space basis

For the activation-subspace intervention, let $A_T \in \mathbb{R}^{n \times d}$ contain the final-residual vectors on the closed addition probe, let $A_{T,c} = H A_T$, and let P_{q+}^0 be the rank-5 centered task projector. Form the task-restricted activation matrix

$$A_{\text{task}} = P_{q+}^0 A_{T,c} \quad (9)$$

and compute its thin SVD

$$A_{\text{task}} = U \Sigma V^\top.$$

If k singular values exceed 10^{-10} times the largest singular value, retain the first $\min(5, k)$ columns of V . All ten evaluated seeds have $k = 5$, so the intervention basis is $B = V_{:,1:5} \in \mathbb{R}^{d \times 5}$ with $B^\top B = I_5$. The retained-energy diagnostic is

$$\frac{\sum_{j=1}^5 \sigma_j^2}{\sum_j \sigma_j^2},$$

which equals one in the archive because $\text{rank}(P_{q+}^0) = 5$ and the restricted matrix has full available rank in every evaluated seed.

Proposition 10 (Equivalence of the activation basis and restricted readout eNTK). *Let $K_U = p A_T A_T^\top$ represent the exact unembedding eNTK on the probe. Then*

$$P_{q+}^0 H K_U H P_{q+}^0 = p A_{\text{task}} A_{\text{task}}^\top = p U \Sigma^2 U^\top.$$

Consequently, U comprises the nonzero sample-space eigenvectors of the task-restricted unembedding eNTK, while V contains their corresponding directions in activation space. For each retained singular triplet (u_j, σ_j, v_j) ,

$$v_j = \sigma_j^{-1} A_{T,c}^\top P_{q+}^0 u_j.$$

Hence, the basis employed by the intervention is the activation-space lift of the centered task modes chosen via the readout eNTK.

Proof. Proposition 14 gives $K_U = p A_T A_T^\top$. Since P_{q+}^0 is centered, $P_{q+}^0 H = H P_{q+}^0 = P_{q+}^0$. Therefore

$$\begin{aligned} P_{q+}^0 H K_U H P_{q+}^0 &= p P_{q+}^0 H A_T A_T^\top H P_{q+}^0 \\ &= p (P_{q+}^0 A_{T,c}) (P_{q+}^0 A_{T,c})^\top \\ &= p A_{\text{task}} A_{\text{task}}^\top. \end{aligned}$$

Substituting $A_{\text{task}} = U \Sigma V^\top$ yields

$$p A_{\text{task}} A_{\text{task}}^\top = p U \Sigma V^\top V \Sigma U^\top = p U \Sigma^2 U^\top.$$

For a nonzero singular value, the transposed singular-vector equation is

$$A_{\text{task}}^\top u_j = \sigma_j v_j.$$

Using $A_{\text{task}}^\top = A_{T,c}^\top P_{q+}^0$ and dividing by σ_j proves the lift formula. $\square$

The same B is applied in the common d -dimensional residual coordinate system at the MLP output and final residual. For a row activation h , guided removal and retention are respectively

$$h_{\text{remove}} = h(I - BB^\top), \quad h_{\text{retain}} = hBB^\top.$$

A random rank-matched control replaces B by the orthonormal factor obtained from a reduced QR decomposition of a Gaussian $d \times 5$ matrix. The exact basis seed for model seed m , stage indicator $j = \mathbf{1}\{\text{final residual}\}$, and replicate $r \in \{0, \dots, 19\}$ is

$$700000 + 10000m + 100j + r.$$

A fresh `torch.Generator(device="cpu")` is manually seeded, then used by `torch.randn(d_model, rank)` before `torch.linalg.qr(..., mode="reduced")`. For fixed (m, j, r) , the same basis is reused across removal and retention and across the two random-control types. For rank-and-norm matching, separately at each intervention stage with held-out activation matrix A_s , set

$$\gamma_s = \frac{\|A_s B B^\top\|_F}{\|A_s B_0 B_0^\top\|_F}.$$

The matched random component is $\gamma_s h B_0 B_0^\top$; it replaces $h B B^\top$ in the removal or retention formula. If the denominator is at most 10^{-30} , the implementation uses $\gamma_s = 1$. This matches the held-out-set Frobenius norm of the manipulated activation component without changing its rank; it is not basis-norm matching. Outcomes are evaluated immediately on the full held-out test table, without parameter updates, using accuracy, mean cross-entropy, and mean correct-logit margin.

The archive contains all 1,640 intervention rows: 800 rank-matched random rows, 800 rank-and-norm-matched random rows, and 40 guided rows over ten model seeds, two stages, and two intervention modes. It also contains a manifest for the 400 unique random bases. The original sampled matrices and generator states are absent, and the original PyTorch version was not recorded. The deterministic procedure and aggregate results are therefore reproducible at the algorithmic level, but bitwise regeneration of the original bases is not claimed.

D Metric Definitions and Supporting Proofs

This appendix collects the formal definitions and proofs behind the symmetry claims. Throughout, $s(a, b) = a + b \pmod{p}$. For each $\sigma \in \mathbb{Z}_p$, let

$$\mathcal{O}_\sigma = \{(a, b) \in \mathbb{Z}_p^2 : a + b = \sigma\}$$

denote the complete sum-orbit with sum σ . A probe set is a union of r complete sum-orbits,

$$S = \bigcup_{j=1}^r \mathcal{O}_{\sigma_j}, \quad \sigma_j \neq \sigma_k \text{ for } j \neq k,$$

so $n = |S| = rp$. For matrices indexed by $S \times S$ we use the Frobenius inner product

$$\langle A, B \rangle_F = \sum_{x, x' \in S} A_{x, x'} B_{x, x'}, \quad \|A\|_F^2 = \langle A, A \rangle_F.$$

The centering matrix is $H = I - n^{-1} \mathbf{1} \mathbf{1}^\top$. We write $\tilde{K} = H K H / \text{Tr}(H K H)$ for a normalized full kernel and $\tilde{K}^{(\ell)} = H K^{(\ell)} H / \text{Tr}(H K^{(\ell)} H)$ for an independently normalized block, whenever the corresponding centered trace is positive.

Lemma 1 (Fourier subspace rank on a closed probe set). *Let $S \subset \mathbb{Z}_p^2$ be r complete sum-orbits. The span of the sum-Fourier characters restricted to S has dimension exactly r , and any addition-stationary kernel on S has rank at most r .*

A kernel K on $\mathbb{Z}_p^2$ is *addition-stationary* if there exists $\kappa : \mathbb{Z}_p \rightarrow \mathbb{R}$ such that

$$K(x, x') = \kappa(s(x) - s(x')).$$

On a restricted probe set S , the corresponding linear subspace contains matrices whose entries depend only on the sum difference $s(x) - s(x')$ among entries present in $S \times S$.

Sum-invariance residual. For $\delta \in \mathbb{Z}_p$, P_δ is the permutation induced by $T_\delta(a, b) = (a + \delta, b - \delta)$. Then

$$R_t^{\text{sum}} = \frac{1}{p} \sum_{\delta} \frac{\|\tilde{K}_t - P_\delta^\top \tilde{K}_t P_\delta\|_F}{\|\tilde{K}_t\|_F}.$$

Addition-stationary error. With Π_{stat} the orthogonal projector onto the addition-stationary subspace (Proposition 11),

$$E_t^{\text{stat}} = \frac{\|\tilde{K}_t - \Pi_{\text{stat}}(\tilde{K}_t)\|_{\text{F}}}{\|\tilde{K}_t\|_{\text{F}}}.$$

Proposition 11 (Orthogonal projector for E^{stat}). *Let S be any probe set and define an equivalence relation on $S \times S$ by $(x, x') \sim (y, y') \iff s(x) - s(x') = s(y) - s(y')$. Let $\mathcal{A}_{\text{stat}}(S) \subseteq \mathbb{R}^{S \times S}$ denote the subspace of matrices constant on each sum-difference equivalence class. The Frobenius-orthogonal projector onto it is given by entrywise blockwise averaging over equivalence classes:*

$$(\Pi_{\text{stat}} A)_{x, x'} = \frac{1}{|[x, x']|} \sum_{(u, v) \in [x, x']} A_{u, v}.$$

Proof. Let $\mathcal{C}$ be the set of equivalence classes induced by $\sim$. The classes are disjoint and cover $S \times S$. For $C \in \mathcal{C}$, write $\mathbf{1}_C \in \mathbb{R}^{S \times S}$ for the matrix indicator of C . Since the classes are disjoint,

$$\left\langle \frac{\mathbf{1}_C}{\sqrt{|C|}}, \frac{\mathbf{1}_{C'}}{\sqrt{|C'|}} \right\rangle_{\text{F}} = \mathbf{1}\{C = C'\}.$$

Thus the normalized class indicators form an orthonormal basis for $\mathcal{A}_{\text{stat}}(S)$: a matrix is in $\mathcal{A}_{\text{stat}}(S)$ exactly when it is a linear combination of these indicators.

The Frobenius-orthogonal projection of A onto this basis is therefore

$$\begin{aligned} \Pi_{\text{stat}} A &= \sum_{C \in \mathcal{C}} \left\langle A, \frac{\mathbf{1}_C}{\sqrt{|C|}} \right\rangle_{\text{F}} \frac{\mathbf{1}_C}{\sqrt{|C|}} \\ &= \sum_{C \in \mathcal{C}} \left(\frac{1}{|C|} \sum_{(u, v) \in C} A_{u, v} \right) \mathbf{1}_C. \end{aligned}$$

Reading off the entry in the class $[(x, x')]$ gives the stated blockwise-average formula. Equivalently, the residual $A - \Pi_{\text{stat}} A$ has zero mean on every class and is therefore orthogonal to every matrix that is constant on classes, which verifies optimality directly. $\square$

Remark 1 (Relation between the two symmetry diagnostics). *Addition-stationarity implies invariance under within-sum translations, because $s(T_\delta x) = s(x)$. Therefore, an exactly addition-stationary kernel has $R^{\text{sum}} = 0$. The converse does not necessarily hold for the specific residual R^{sum} employed here, which averages conjugation by the diagonal action $K \mapsto P_\delta^\top K P_\delta$; a kernel may be invariant under this diagonal action while still exhibiting non-stationary dependence on the two sums. For this reason, R^{sum} and E^{stat} are presented as complementary checks rather than as interchangeable measures.*

D.1 Proof of Lemma 1

Proof. Write $S = \bigcup_{j=1}^r \mathcal{O}_{\sigma_j}$ with distinct $\sigma_j \in \mathbb{Z}_p$. A sum-Fourier character

$$\psi_\omega(a, b) = \exp(2\pi i \omega(a + b)/p)$$

is constant on each orbit $\mathcal{O}_{\sigma_j}$ and takes value $\exp(2\pi i \omega \sigma_j/p)$ there. Therefore every restricted sum-Fourier function is orbit-constant on S , and the restricted span has dimension at most the number of orbits, r .

It remains to show that the upper bound is attained. Let $\alpha_j = \exp(2\pi i \sigma_j/p)$. The $r \times r$ matrix formed from frequencies $\omega = 0, \dots, r-1$ is

$$V_{j\omega} = \exp(2\pi i \omega \sigma_j/p) = \alpha_j^\omega.$$

This is a Vandermonde matrix. Since the sums σ_j are distinct modulo p , the roots α_j are distinct, and

$$\det(V) = \prod_{1 \leq j < k \leq r} (\alpha_k - \alpha_j) \neq 0.$$

Hence the restricted characters with frequencies $0, \dots, r-1$ are linearly independent on the r selected orbit labels. The complex restricted span has dimension exactly r . The real span used by the metrics has the same dimension: it is the real vector space of all real-valued functions that are constant on the r orbits, equivalently the real and imaginary parts of the complex Fourier span restricted to S .

For the rank claim, let $B \in \{0, 1\}^{|S| \times r}$ be the orbit-incidence matrix, $B_{xj} = \mathbf{1}\{x \in \mathcal{O}_{\sigma_j}\}$. If K is addition-stationary on S , then for $x \in \mathcal{O}_{\sigma_j}$ and $x' \in \mathcal{O}_{\sigma_k}$,

$$K(x, x') = \kappa(\sigma_j - \sigma_k).$$

Thus $K = BCB^\top$ where $C_{jk} = \kappa(\sigma_j - \sigma_k)$. Since the columns of B have disjoint nonempty supports, $\text{rank}(B) = r$, and therefore

$$\text{rank}(K) \leq \min\{\text{rank}(B), \text{rank}(C), \text{rank}(B^\top)\} \leq r.$$

□

Remark 2 (Interpretation on the restricted probe set). *The proof demonstrates that, after restricting to the r selected sums, the span of all sum-Fourier characters coincides exactly with the space of functions constant on each sum-orbit. Therefore, P_{add} tests sum-orbit dependence on the sums examined; by itself it does not isolate a sparse set of global Fourier frequencies nor differentiate a trigonometric representation from another function of the sum.*

Proposition 12 (Spectral structure of addition-stationary kernels). *Let K be addition-stationary on $\mathbb{Z}_p^2$. Then (i) K is invariant under T_δ ; (ii) $\psi_\omega(a, b) = \exp(2\pi i \omega(a + b)/p)$ are eigenfunctions of the integral operator induced by K ; (iii) the full-grid kernel matrix has rank at most p , dropping to r on a probe set of r orbits.*

Proof. First, $s(T_\delta x) = s(x)$ for every $\delta \in \mathbb{Z}_p$. Therefore

$$\begin{aligned} K(T_\delta x, T_\eta x') &= \kappa(s(T_\delta x) - s(T_\eta x')) \\ &= \kappa(s(x) - s(x')) \\ &= K(x, x') \end{aligned}$$

for any within-orbit shifts δ, η . This proves the stated invariance, and in fact a slightly stronger independent invariance in each argument.

For the spectral claim, consider the unnormalized finite-grid integral operator

$$(\mathcal{K}g)(x) = \sum_{x' \in \mathbb{Z}_p^2} K(x, x')g(x').$$

For $\psi_\omega(x) = \exp(2\pi i \omega s(x)/p)$, group the summation by the sum value $t = s(x')$. There are exactly p pairs $x' \in \mathbb{Z}_p^2$ with sum t . Hence

$$\begin{aligned} (\mathcal{K}\psi_\omega)(x) &= p \sum_{t \in \mathbb{Z}_p} \kappa(s(x) - t) \exp(2\pi i \omega t/p) \\ &= p \exp(2\pi i \omega s(x)/p) \\ &\quad \cdot \sum_{u \in \mathbb{Z}_p} \kappa(u) \exp(-2\pi i \omega u/p) \\ &= \lambda_\omega \psi_\omega(x), \end{aligned}$$

where

$$\lambda_\omega = p \sum_{u \in \mathbb{Z}_p} \kappa(u) \exp(-2\pi i \omega u/p).$$

Thus the sum-Fourier characters diagonalize every full-grid addition-stationary kernel. If the operator is normalized by $1/p^2$, the same eigenfunctions appear and all eigenvalues are scaled by $1/p^2$.

Finally, let $M \in \{0, 1\}^{p^2 \times p}$ be the full-grid sum-incidence matrix, $M_{x\sigma} = \mathbf{1}\{s(x) = \sigma\}$, and let $C_{\sigma\tau} = \kappa(\sigma - \tau)$. Then the full-grid kernel matrix factors as $K = MCM^\top$, so $\text{rank}(K) \leq p$. Restricting to a probe set of r complete orbits replaces M by the $|S| \times r$ incidence matrix in Lemma 1, giving rank at most r . □

Proposition 13 (Centered-kernel ceiling and isotropic baselines). *Let P_{add} be the orthogonal projector onto the restricted orbit-constant subspace (equivalently, the restricted span of all sum-Fourier characters) on a probe set of r complete sum-orbits. Then: (i) $\mathbf{1} \in \text{im}(P_{\text{add}})$; (ii) for any rank- m orthogonal projector Q with $\text{im}(Q) \subseteq \mathbf{1}^\perp$,*

$$\text{Tr}(P_{\text{add}}Q) \leq \min\{m, r-1\};$$

(iii) for a rotationally invariant random rank- m projector Q on $\mathbb{R}^n$,

$$\mathbb{E}[m^{-1}\text{Tr}(P_{\text{add}}Q)] = r/n;$$

and (iv) for a rotationally invariant random rank- m projector Q_0 inside the centered subspace $\mathbf{1}^\perp$,

$$\mathbb{E}[m^{-1}\text{Tr}(P_{\text{add}}Q_0)] = (r-1)/(n-1).$$

In particular, $m = r$ exact recovery on a centered kernel gives raw overlap $(r-1)/r = 5/6$ rather than 1.

Proof. The frequency-0 sum-Fourier character is constant on S , so $\mathbf{1} \in \text{im}(P_{\text{add}})$. Since $\text{rank}(P_{\text{add}}) = r$, the centered part of the orbit-constant subspace is

$$\mathcal{F}_{\text{add}}^0 = \text{im}(P_{\text{add}}) \cap \mathbf{1}^\perp, \quad \dim(\mathcal{F}_{\text{add}}^0) = r-1.$$

Equivalently, with $u = \mathbf{1}/\sqrt{n}$ and $H = I - uu^\top$,

$$P_{\text{add}} = uu^\top + P_{\text{add}}^0,$$

where P_{add}^0 is the orthogonal projector onto $\mathcal{F}_{\text{add}}^0$.

If Q is an orthogonal projector with $\text{im}(Q) \subseteq \mathbf{1}^\perp$, then $uu^\top Q = 0$ and

$$\text{Tr}(P_{\text{add}}Q) = \text{Tr}(P_{\text{add}}^0Q).$$

For two orthogonal projectors, $\text{Tr}(P_{\text{add}}^0Q)$ is the sum of squared cosines of principal angles between their ranges, so it is at most the smaller of their ranks. Hence

$$\begin{aligned} \text{Tr}(P_{\text{add}}Q) &\leq \min\{\text{rank}(P_{\text{add}}^0), \text{rank}(Q)\} \\ &= \min\{r-1, m\}. \end{aligned}$$

This proves the centered ceiling.

For the full-space isotropic baseline, rotational invariance on the Grassmannian implies $\mathbb{E}[Q]$ commutes with every orthogonal matrix. By Schur's lemma, or by the elementary fact that the only matrices commuting with all rotations are scalar multiples of the identity, $\mathbb{E}[Q] = \alpha I_n$. Taking traces gives mn^{-1} for the scalar: $m = \mathbb{E}\text{Tr}(Q) = \alpha n$, so $\alpha = m/n$. Therefore

$$\begin{aligned} \mathbb{E}[m^{-1}\text{Tr}(P_{\text{add}}Q)] &= m^{-1}\text{Tr}(P_{\text{add}}\mathbb{E}Q) \\ &= m^{-1}\text{Tr}(P_{\text{add}}(m/n)I_n) \\ &= r/n. \end{aligned}$$

For the centered isotropic baseline, the same argument within the $(n-1)$ -dimensional subspace $\mathbf{1}^\perp$ gives

$$\mathbb{E}[Q_0] = \frac{m}{n-1}H.$$

Since $\text{Tr}(P_{\text{add}}H) = \text{Tr}(P_{\text{add}}) - \text{Tr}(P_{\text{add}}uu^\top) = r-1$, we obtain

$$\mathbb{E}[m^{-1}\text{Tr}(P_{\text{add}}Q_0)] = \frac{r-1}{n-1}.$$

The score in Eq. 2 uses the full-space baseline r/n as a fixed reporting convention inherited from uncentered random-overlap normalization; the centered baseline is displayed here to make explicit the small null-model shift caused by centering. The empirical comparisons in the paper are blockwise and use the same convention for every block. $\square$

E Scalarized Form of Lin’s Unembedding eNTK Identity

Lin [2025, Eq. 19] derives the per-class unembedding kernel as a class-diagonal residual Gram matrix. The following proposition restates this result after summing over output classes in the scalarized convention adopted in this paper and describes the impact of centering and trace-normalization.

Proposition 14 (Scalarized form of Lin’s unembedding identity). *Let $W_U \in \mathbb{R}^{d_{\text{model}} \times p}$ be the unembedding matrix, $h_T(x) \in \mathbb{R}^{d_{\text{model}}}$ the final residual-stream vector at the = position, and $f_c(x; \theta) = W_U[:, c]^\top h_T(x)$. In the blockwise eNTK contribution for the unembedding block, $h_T(x)$ is treated as fixed when differentiating with respect to W_U because all upstream parameters are outside the block. Then*

$$\begin{aligned} K_t^{(\text{unembed})}(x, x') &= \sum_{c=1}^p \langle \nabla_{W_U} f_c(x), \nabla_{W_U} f_c(x') \rangle_{\text{F}} \\ &= p \langle h_T(x), h_T(x') \rangle. \end{aligned}$$

Consequently, if $G_t(x, x') = \langle h_T(x), h_T(x') \rangle$ and $\text{Tr}(HG_tH) > 0$, then

$$\hat{K}_t^{(\text{unembed})} = \frac{H K_t^{(\text{unembed})} H}{\text{Tr}(H K_t^{(\text{unembed})} H)} = \frac{H G_t H}{\text{Tr}(H G_t H)}.$$

Proof. Fix the training time t and suppress it from notation. Let $e_c \in \mathbb{R}^p$ be the c -th standard basis vector. The logit can be written as

$$f_c(x) = h_T(x)^\top W_U e_c.$$

For an entry $W_U[i, j]$,

$$\frac{\partial f_c(x)}{\partial W_U[i, j]} = h_{T,i}(x) e_c[j] = h_{T,i}(x) \mathbf{1}\{j = c\}.$$

Thus the full matrix-valued gradient is the rank-one matrix

$$\nabla_{W_U} f_c(x) = h_T(x) e_c^\top.$$

The Frobenius inner product between the two class- c gradients is

$$\begin{aligned} \langle h_T(x) e_c^\top, h_T(x') e_c^\top \rangle_{\text{F}} &= \text{Tr}((h_T(x) e_c^\top)^\top h_T(x') e_c^\top) \\ &= \text{Tr}(e_c h_T(x)^\top h_T(x') e_c^\top) \\ &= \langle h_T(x), h_T(x') \rangle \text{Tr}(e_c e_c^\top) \\ &= \langle h_T(x), h_T(x') \rangle. \end{aligned}$$

Equivalently, vectorizing the gradients gives

$$\text{vec}(\nabla_{W_U} f_c(x)) = e_c \otimes h_T(x),$$

so the same inner product follows from

$$\langle e_c \otimes h_T(x), e_c \otimes h_T(x') \rangle = \langle e_c, e_c \rangle \langle h_T(x), h_T(x') \rangle.$$

Summing the scalarized eNTK over the p output logits gives

$$K^{(\text{unembed})}(x, x') = \sum_{c=1}^p \langle h_T(x), h_T(x') \rangle = p G(x, x'),$$

where $G(x, x') = \langle h_T(x), h_T(x') \rangle$.

The normalized identity follows by homogeneity. Since $K^{(\text{unembed})} = pG$,

$$\begin{aligned} H K^{(\text{unembed})} H &= p H G H, \\ \text{Tr}(H K^{(\text{unembed})} H) &= p \text{Tr}(H G H). \end{aligned}$$

When the centered residual Gram matrix is nonzero, the factor p cancels in trace-normalization, yielding the claimed equality of centered, normalized matrices. No assumption about training dynamics, loss, optimizer, or grokking is used; the result is purely an architectural identity for a final linear readout. $\square$

Remark 3 (Biases, class averaging, and estimator conventions). *The proposition matches the scalarized eNTK used in the paper, which sums over logits. If the implementation averages rather than sums over logits, the raw constant becomes 1 instead of p ; the centered, trace-normalized identity is unchanged. If the final readout included an unembedding bias $b_U \in \mathbb{R}^p$, then $\nabla_{b_U} f_c = e_c$ would contribute an additional constant kernel $\sum_c \langle e_c, e_c \rangle = p$. This term is annihilated by centering, since $H11^\top H = 0$, so the centered normalized identity would still be unchanged provided the residual Gram contribution has positive centered trace. Our reported architecture has no unembedding bias.*

F When Fourier Residual Streams Yield Addition-Stationary Gram Matrices

Proposition 15 (Fourier residual stream implies addition-stationary Gram matrix). *Suppose*

$$h_T(x) = \sum_{\omega \in \mathcal{K}} (u_\omega \cos(\frac{2\pi\omega s(x)}{p}) + v_\omega \sin(\frac{2\pi\omega s(x)}{p})),$$

where $\mathcal{K} \subseteq \{1, \dots, p-1\}$, the feature directions satisfy

$$\langle u_\omega, u_{\omega'} \rangle = \langle v_\omega, v_{\omega'} \rangle = \langle u_\omega, v_{\omega'} \rangle = 0 \quad \text{for } \omega \neq \omega',$$

and for each $\omega \in \mathcal{K}$,

$$\langle u_\omega, v_\omega \rangle = 0, \quad \|u_\omega\|^2 = \|v_\omega\|^2 = \alpha_\omega.$$

Then

$$\langle h_T(x), h_T(x') \rangle = \sum_{\omega \in \mathcal{K}} \alpha_\omega \cos(\frac{2\pi\omega(s(x) - s(x'))}{p}),$$

which is addition-stationary.

Proof. Let

$$\phi_\omega(x) = \frac{2\pi\omega s(x)}{p}.$$

For readability, write $C_\omega = \cos \phi_\omega(x)$, $S_\omega = \sin \phi_\omega(x)$, $C'_\omega = \cos \phi_\omega(x')$, and $S'_\omega = \sin \phi_\omega(x')$. Expanding the Gram inner product gives

$$\begin{aligned} \langle h_T(x), h_T(x') \rangle &= \sum_{\omega, \omega' \in \mathcal{K}} \langle u_\omega, u_{\omega'} \rangle C_\omega C'_{\omega'} \\ &\quad + \sum_{\omega, \omega' \in \mathcal{K}} \langle u_\omega, v_{\omega'} \rangle C_\omega S'_{\omega'} \\ &\quad + \sum_{\omega, \omega' \in \mathcal{K}} \langle v_\omega, u_{\omega'} \rangle S_\omega C'_{\omega'} \\ &\quad + \sum_{\omega, \omega' \in \mathcal{K}} \langle v_\omega, v_{\omega'} \rangle S_\omega S'_{\omega'}. \end{aligned}$$

All cross-frequency terms vanish by assumption. For each fixed ω , the mixed sine-cosine terms vanish because $\langle u_\omega, v_\omega \rangle = 0$, while the remaining cosine-cosine and sine-sine terms have the same coefficient α_ω . Hence

$$\begin{aligned} \langle h_T(x), h_T(x') \rangle &= \sum_{\omega \in \mathcal{K}} \alpha_\omega [C_\omega C'_\omega + S_\omega S'_\omega] \\ &= \sum_{\omega \in \mathcal{K}} \alpha_\omega \cos(\phi_\omega(x) - \phi_\omega(x')) \\ &= \sum_{\omega \in \mathcal{K}} \alpha_\omega \cos\left(\frac{2\pi\omega(s(x) - s(x'))}{p}\right), \end{aligned}$$

using $\cos A \cos B + \sin A \sin B = \cos(A - B)$. The final expression depends only on $s(x) - s(x')$, so it is addition-stationary with

$$\kappa(\Delta) = \sum_{\omega \in \mathcal{K}} \alpha_\omega \cos(2\pi\omega\Delta/p).$$

□

Remark 4 (Approximate version). *The exact proposition is a sufficient condition, not a necessary one. A useful perturbative form is obtained by writing, for each frequency,*

$$a_\omega = \|u_\omega\|^2, \quad b_\omega = \|v_\omega\|^2, \quad c_\omega = \langle u_\omega, v_\omega \rangle.$$

The same-frequency contribution equals

$$\begin{aligned} & a_\omega \cos \phi_\omega(x) \cos \phi_\omega(x') + b_\omega \sin \phi_\omega(x) \sin \phi_\omega(x') \\ & + c_\omega [\cos \phi_\omega(x) \sin \phi_\omega(x') + \sin \phi_\omega(x) \cos \phi_\omega(x')] \\ & = \frac{a_\omega + b_\omega}{2} \cos(\phi_\omega(x) - \phi_\omega(x')) \\ & + \frac{a_\omega - b_\omega}{2} \cos(\phi_\omega(x) + \phi_\omega(x')) \\ & + c_\omega \sin(\phi_\omega(x) + \phi_\omega(x')). \end{aligned}$$

The first term is addition-stationary; the latter two are not. Cross-frequency inner products add further non-stationary products involving $\phi_\omega(x)$ and $\phi_{\omega'}(x')$. Thus the deviation from addition-stationarity is controlled by sine/cosine norm imbalance $|a_\omega - b_\omega|$, within-frequency non-orthogonality $|c_\omega|$, and cross-frequency inner products. This is the formal caveat behind the main-text wording that a Fourier residual stream yields an approximately addition-stationary Gram matrix when those additional quantities are small.

G Additional Experimental Protocol Details

Choice of prime moduli. Prime moduli correspond to the standard modular-addition setting and prevent divisor-specific Fourier structure. The unembedding identity itself does not depend on primality.

Endpoint smoothing and retrospective trigger. Final block values average the last four eNTK checkpoints for each run budget. In the dense $p = 113$ reproduction, alignment is measured every 100 epochs and accuracy every 50 epochs. For seed i and metric M , let M_i^{base} be the mean of the first four pre-grok checkpoints, let M_i^{end} be a post-grok endpoint, and define $d_i = M_i^{\text{end}} - M_i^{\text{base}}$ and $s_i = \text{sign}(d_i)$. The retrospective trigger is

$$\begin{aligned} t_i^{20}(M) &= \min\{t < t_{\text{grok},i} : \\ & s_i[M_i(t) - M_i^{\text{base}}] \geq 0.2|d_i|\}. \end{aligned}$$

The strict endpoint averages all available checkpoints after $t_{\text{grok},i}$; the last-four-post-grok endpoint is a sensitivity check. Both depend on future information and are therefore retrospective.

$m = 5$ vs. $m = 6$. For each block and checkpoint, calculate the top- m eigenspace of the block kernel that has been independently centered and trace-normalized. Use Eq. 2 for both $m \in \{5, 6\}$.

Blockwise R^{sum} and E^{stat} . For each block ℓ , $\hat{K}^{(\ell)} = HK^{(\ell)}H/\text{Tr}(HK^{(\ell)}H)$, then apply the same definitions as for the full eNTK.

Freeze- W_U intervention. From the last checkpoint strictly before t_{grok} , create three continuations: matched resume, freeze W_U , and matched-size embedding-block freeze (W_E). Train each to 9,000 epochs.

Reinitialize- W_U intervention. From the first checkpoint after t_{grok} , create matched resume and reinit- W_U continuations. Reinit draws W_U from the original initialization distribution. Recovery is defined by the same recorded-evaluation rule as grokking: the first evaluation in two consecutive evaluation checkpoints above 95%.

Data and compute. All datasets consist of synthetic modular-arithmetic tables produced by the experiment code using the fixed split described earlier. Each run employs one GPU per process. The original experiments utilized NVIDIA RTX 6000 Pro and H100 GPUs, and a verified reproduction used an A100-SXM4-80GB. Table 4 presents observed aggregate traces along with reconstructed estimates for training, eNTK evaluation, peak memory, archived workload, and total project compute, including plausible ranges when direct telemetry was unavailable.

Code and data availability. Code, machine-readable protocols, deterministic seed rules, all reported per-seed result tables, draw-level causal controls, statistical and plotting scripts, checksums, and reproduction instructions are provided in the repository at <https://anonymous.4open.science/r/entk-grokking-5F13>. These materials reproduce the primary reported numerical analyses and figures. Full retraining of the six-modulus sweep and some auxiliary figure writers are not included; original checkpoints, raw eNTK tensors, sampled random bases, and a bitwise lock of the original software environment are also unavailable, so those more limited forms of reconstruction are not claimed.

H Additional Robustness Results

H.1 Centered-rank sensitivity: five versus six modes

Because centered kernels eliminate the constant sum mode, the $m = 6$ score in Eq. 2 contains one dimension that a centered eigenvector cannot occupy. To exclude an artifact from this reporting convention, we recompute all per-block alignments with $m = 5$, which directly targets the five informative nonconstant sum modes (Table 11).

The $m = 5$ analysis preserves the unembedding-dominant result and in fact strengthens it: the grokked-by-budget unembedding median rises from $+0.645$ ($m = 6$) to $+0.752$ ($m = 5$), the not-grokked-by-budget median changes from $+0.151$ to $+0.166$, and the grokked-by-budget vs. not-grokked-by-budget landmark Mann–Whitney p remains strongly significant at 0.0002. The other three blocks change negligibly. The increase in unembedding values under $m = 5$ is consistent with Proposition 13: the constant mode is unrecoverable in centered coordinates, so removing it raises the attainable overlap. This ceiling argument does not predict the numerical increase exactly because the estimated top- m eigenspace also changes with m .

H.2 Blockwise sum-invariance and stationary error

The alignment score evaluates whether the leading eigenspace overlaps with the restricted orbit-constant subspace. The invariance metrics address the complementary issue: whether the entire normalized block kernel becomes invariant to within-sum translations and approximately stationary along the sum coordinate. We compute R^{sum} and E^{stat} separately for each independently centered and trace-normalized block eNTK. Table 12 presents these values in a per-modulus format for the completed $p = 59$, $p = 83$, $p = 89$, $p = 101$, $p = 113$, and $p = 127$ runs.

The $p = 59$, $p = 83$, $p = 89$, and $p = 101$ rows maintain the same qualitative ordering as the alignment results. At $p = 59$, the unembedding shows a 34.2% reduction in R^{sum} and a 31.8% reduction in E^{stat} ; at $p = 83$, the corresponding reductions are 16.3% and 17.4%; at $p = 89$, they are 27.2% and 24.7%; at $p = 101$, they are 6.8% and 3.1%. The $p = 113$ row is mixed: attention has the largest R^{sum} reduction (10.7%), whereas unembedding has the largest E^{stat} reduction (4.7%). The $p = 127$ row is qualitatively different: the unembedding increases on both metrics (-14.2% and -7.2% reductions), while embed shows the largest R^{sum} reduction and attention shows the largest E^{stat} reduction. These metrics do not depend on the alignment score’s eigenspace cutoff and apply to the entire kernel, so their agreement or disagreement provides information rather than reiterating the alignment result.

H.3 Numerical verification of the gradient identity

Following the per-class identity of Lin [2025], Section 5 and Proposition 14 restate the scalarized relation $K_t^{(\text{unembed})} = p \cdot G_t$, where $G_t(x, x') = \langle h_T(x), h_T(x') \rangle$ is the residual-stream Gram matrix. After centering and trace-normalization, the constant p cancels and $\hat{K}_t^{(\text{unembed})}$ should equal the centered, normalized Gram matrix exactly. We verify this on a final-epoch checkpoint of seed 0: comparing the random-projection eNTK (with $R = 64$) to the closed-form Gram matrix gives a relative Frobenius error of 2.88×10^{-6} and a max element-wise error of 1.81×10^{-8} . This supports numerical agreement to relative error of order 10^{-6} .

The exact identity supplies an estimator check, but the observed discrepancy alone does not establish its numerical source. Under a projection scheme whose class-space Rademacher vectors have fixed

Table 11: **Centered-rank sensitivity (medians).** Final per-block $\bar{A}^{\text{add}}$ medians by modulus under $m = 6$ (centered, including the unrecoverable constant mode) and $m = 5$ (informative nonconstant modes). The unembedding-dominant result remains unchanged.

p	m	split	embed	attn	mlp	unembed
59	6	grokked by budget	−0.014	+0.029	−0.010	+0.603
	6	not grokked by budget	−0.012	−0.006	−0.014	+0.273
	5	grokked by budget	−0.014	+0.026	−0.011	+0.725
	5	not grokked by budget	−0.012	−0.007	−0.015	+0.300
83	6	grokked by budget	−0.011	+0.034	−0.008	+0.719
	6	not grokked by budget	−0.008	−0.004	−0.010	+0.251
	5	grokked by budget	−0.011	+0.015	−0.010	+0.735
	5	not grokked by budget	−0.008	−0.004	−0.011	+0.301
89	6	grokked by budget	−0.010	+0.035	+0.040	+0.642
	6	not grokked by budget	−0.008	−0.005	−0.010	+0.394
	5	grokked by budget	−0.011	+0.039	−0.008	+0.694
	5	not grokked by budget	−0.008	−0.005	−0.010	+0.451
101	6	grokked by budget	−0.009	+0.027	+0.001	+0.686
	6	not grokked by budget	−0.007	−0.004	−0.009	+0.311
	5	grokked by budget	−0.009	+0.024	−0.008	+0.724
	5	not grokked by budget	−0.007	−0.005	−0.009	+0.354
113	6	grokked by budget	−0.008	−0.001	−0.007	+0.645
	6	not grokked by budget	−0.006	−0.004	−0.008	+0.151
	5	grokked by budget	−0.008	−0.002	−0.008	+0.752
	5	not grokked by budget	−0.007	−0.004	−0.008	+0.166
127	6	grokked by budget	−0.007	−0.006	−0.006	+0.532
	6	not grokked by budget	−0.006	−0.005	−0.007	+0.286
	5	grokked by budget	−0.007	−0.006	−0.007	+0.621
	5	not grokked by budget	−0.006	−0.005	−0.007	+0.318

squared norm and the stated scaling, the unembedding estimate is algebraically projection-noise-free because $\nabla_{W_U} f_c = h_T(x) \otimes e_c$ factors into the residual vector and a class direction. The implementation should record its dtype and projection scaling when reproducing this check.

H.4 Projection-count robustness across blocks

Under the estimator scaling described above, the gradient identity predicts zero projection variance specifically for the unembedding block. The other blocks lack this factorization, so their alignment values could in principle shift with R . Table 14 shows that the reported medians remain stable across $R \in \{16, 32, 64\}$ for the completed projection-count checks ($p = 113$, $p = 59$, $p = 83$, $p = 89$, $p = 101$, and $p = 127$). The block ranking and the conclusion that the unembedding block dominates remain unchanged.

The $p = 83$, $p = 89$, $p = 101$, and $p = 127$ rows report medians over checkpoints crossed with grokked seeds; the $p = 113$ unembedding median is stable at 0.647 across projection counts.

Table 12: **Blockwise sum-invariance and stationary error by modulus.** Lower values indicate better performance. Bold entries denote the largest reduction within each metric-modulus block. MW p values represent comparisons between grokked-by-budget endpoint medians and not-grokked-by-budget final medians, reported either exactly or as conservative power-of-ten upper bounds. A stricter post-grok evaluation should consider only checkpoints after t_{grok} .

p	Metric	Block	pre-grok med	endpoint med	reduction %	NGB final med	MW p
59	R^{sum}	embed	1.043	0.987	5.4	1.104	$< 10^{-3}$
	R^{sum}	attn	1.143	1.137	0.6	1.213	$< 10^{-3}$
	R^{sum}	mlp	0.991	0.984	0.7	1.074	$< 10^{-3}$
	R^{sum}	unembed	0.920	0.605	34.2	1.158	$< 10^{-3}$
	E^{stat}	embed	1.000	0.999	0.0	0.999	$< 10^0$
	E^{stat}	attn	0.985	0.978	0.7	0.998	$< 10^{-3}$
	E^{stat}	mlp	0.990	0.987	0.3	0.997	$< 10^{-3}$
	E^{stat}	unembed	0.735	0.502	31.8	0.903	$< 10^{-3}$
	R^{sum}	embed	1.070	1.044	2.4	1.178	0.0008
	R^{sum}	attn	1.192	1.126	5.5	1.241	0.0058
	R^{sum}	mlp	0.980	0.974	0.6	1.139	0.0017
	R^{sum}	unembed	0.949	0.794	16.3	1.158	0.0010
83	E^{stat}	embed	1.000	1.000	0.0	1.000	0.0073
	E^{stat}	attn	0.995	0.983	1.3	0.999	0.0003
	E^{stat}	mlp	0.992	0.990	0.2	0.998	0.0003
	E^{stat}	unembed	0.785	0.648	17.4	0.888	0.0013
	R^{sum}	embed	1.054	1.012	4.0	1.094	$< 10^{-3}$
	R^{sum}	attn	1.186	1.087	8.3	1.232	$< 10^{-4}$
	R^{sum}	mlp	0.984	0.979	0.5	1.046	$< 10^{-4}$
	R^{sum}	unembed	0.970	0.706	27.2	1.036	$< 10^{-4}$
	E^{stat}	embed	1.000	1.000	0.0	1.000	$< 10^{-1}$
	E^{stat}	attn	0.996	0.983	1.3	0.999	$< 10^{-4}$
	E^{stat}	mlp	0.993	0.990	0.4	0.996	$< 10^{-4}$
	E^{stat}	unembed	0.794	0.598	24.7	0.799	$< 10^{-4}$

H.5 Additional-modulus scope check

We repeat the experiment at additional moduli with the same baseline recipe, changing the epoch budget where needed to expose late grokking (Table 15). At $p = 59$, 10 of 20 seeds grok under a 15,000-epoch budget, with median grok time 13,425 and a final unembedding–accuracy Spearman correlation of $\rho = 0.868$ ($p = 6.94 \times 10^{-7}$). The $p = 83$ run has 10 grokked seeds at 8,500 epochs, median grok time 6,700, and Spearman $\rho = 0.902$ ($p = 5.61 \times 10^{-8}$). For $p = 89$, 12 seeds grok within 10,000 epochs, median grok time is 8,100, and the final unembedding–accuracy correlation is $\rho = 0.926$ ($p = 4.94 \times 10^{-9}$). The $p = 101$ row gives 12 grokked seeds, median grok time 6,150, and $\rho = 0.802$ ($p = 2.14 \times 10^{-5}$); the $p = 127$ row gives 9 grokked seeds, median grok time 4,550, and $\rho = 0.892$ ($p = 1.31 \times 10^{-7}$). These additional moduli produce a richer continuum of partial generalization: test accuracies span $[0.051, 1.0]$ at $p = 89$, $[0.095, 1.0]$ at $p = 83$, $[0.052, 0.999]$ at $p = 59$, $[0.049, 1.0]$ at $p = 101$, and $[0.035, 1.0]$ at $p = 127$, while the unembedding alignment tracks this continuum smoothly. The within-block positive-change ratio gives unembed 69.0%, 72.0%, 68.5%, 72.4%, and 99.4% for $p = 59, 83, 89, 101$, and 127, respectively; in every completed modulus, it is the largest ratio.

The scope check indicates that the normalized unembedding diagnostic recurs across the examined moduli, and the dose-response monotonicity becomes more evident in environments with many

Table 13: **Blockwise sum-invariance and stationary error by modulus (continued)**. Lower is better; notation and comparisons follow Table 12.

p	Metric	Block	pre-grok med	endpoint med	reduction %	NGB final med	MW p
101	R^{sum}	embed	1.071	1.026	4.2	1.111	0.0001
	R^{sum}	attn	1.147	1.117	2.6	1.203	0.0030
	R^{sum}	mlp	0.984	0.977	0.7	1.096	0.0001
	R^{sum}	unembed	0.803	0.749	6.8	1.106	0.0001
	E^{stat}	embed	1.000	1.000	−0.0	1.000	0.4727
	E^{stat}	attn	0.996	0.982	1.4	0.999	0.0000
	E^{stat}	mlp	0.991	0.988	0.2	0.998	0.0000
	E^{stat}	unembed	0.625	0.606	3.1	0.853	0.0000
	R^{sum}	embed	1.066	0.971	8.9	1.189	$< 10^{-3}$
	R^{sum}	attn	1.166	1.041	10.7	1.174	$< 10^{-1}$
	R^{sum}	mlp	0.986	0.984	0.1	1.227	$< 10^{-2}$
	R^{sum}	unembed	0.766	0.700	8.7	1.284	$< 10^{-3}$
	E^{stat}	embed	1.000	1.000	0.0	1.000	$< 10^{-1}$
	E^{stat}	attn	0.996	0.985	1.1	0.999	$< 10^{-3}$
	E^{stat}	mlp	0.993	0.991	0.2	0.999	$< 10^{-3}$
	E^{stat}	unembed	0.608	0.579	4.7	0.966	$< 10^{-2}$
127	R^{sum}	embed	1.046	0.999	4.5	1.135	$< 10^{-2}$
	R^{sum}	attn	1.067	1.089	−2.1	1.178	$< 10^{-3}$
	R^{sum}	mlp	1.009	1.005	0.4	1.137	$< 10^{-3}$
	R^{sum}	unembed	0.792	0.904	−14.2	1.103	$< 10^{-2}$
	E^{stat}	embed	1.000	1.000	0.0	1.000	$< 10^0$
	E^{stat}	attn	0.996	0.991	0.5	0.998	$< 10^{-3}$
	E^{stat}	mlp	0.995	0.992	0.2	0.998	$< 10^{-2}$
	E^{stat}	unembed	0.634	0.679	−7.2	0.857	$< 10^{-2}$

partially generalized seeds. The MLP’s within-block positive-change ratio differs across moduli; the unembedding remains the highest ratio in each completed row.

I Summary Tables

J Per-Seed Final Blockwise Alignment Values

Table 18: **Per-seed final per-block $\bar{A}^{\text{add}}$ values for $m = 6$** . Last-4-checkpoint means by modulus ($p = 59$: epochs 13,500–15,000; $p = 83$: epochs 7,000–8,500; $p = 89$: epochs 8,500–10,000; $p = 101$: epochs 6,000–7,500; $p = 113$: epochs 5,500–7,000; $p = 127$: epochs 4,000–5,500). For runs not grokked by budget, t_{grok} is right-censored and shown as undefined.

p	seed	test	t_{grok}	embed	attn	mlp	unembed	grok
59	0	0.996	13400	−0.014	+0.143	+0.145	+0.631	Y
	1	0.113	−	−0.013	−0.005	−0.016	+0.046	N
	2	0.999	8500	−0.016	+0.160	+0.153	+0.484	Y
	3	0.110	−	−0.009	−0.011	−0.015	+0.062	N
	4	0.158	−	−0.012	−0.007	−0.015	+0.195	N
	5	0.052	−	−0.011	−0.010	−0.016	−0.003	N
	6	0.998	10300	−0.015	+0.072	−0.010	+0.630	Y
	7	0.998	13450	−0.014	+0.044	−0.007	+0.606	Y
	8	0.387	−	−0.013	−0.000	−0.001	+0.419	N

Continued in next column

Table 14: **Projection-estimator robustness.** Per-block $\bar{A}^{\text{add}}$ medians by modulus at three projection counts; the $p = 83$, $p = 89$, $p = 101$, and $p = 127$ rows present medians computed over checkpoints crossed with grokked seeds. The unembedding median matches to three decimals across all three R values within each modulus, consistent with the prediction when the estimator applies the fixed-norm class-space projection scaling described above.

p	R	embed	attn	mlp	unembed
59	16	-0.015	+0.051	-0.009	+0.595
	32	-0.015	+0.048	-0.010	+0.595
	64	-0.015	+0.043	-0.009	+0.595
83	16	-0.011	-0.002	-0.008	+0.657
	32	-0.011	-0.001	-0.008	+0.657
	64	-0.011	-0.002	-0.008	+0.657
89	16	-0.010	+0.003	+0.003	+0.619
	32	-0.011	+0.003	+0.004	+0.619
	64	-0.011	+0.008	+0.004	+0.619
101	16	-0.009	+0.007	-0.008	+0.654
	32	-0.009	-0.000	-0.008	+0.654
	64	-0.009	+0.003	-0.008	+0.654
113	16	-0.008	-0.003	-0.008	+0.647
	32	-0.008	-0.004	-0.008	+0.647
	64	-0.008	-0.003	-0.008	+0.647
127	16	-0.007	-0.006	-0.007	+0.608
	32	-0.007	-0.006	-0.007	+0.608
	64	-0.007	-0.005	-0.007	+0.608

Table 15: **Scope check.** The normalized unembedding diagnostic persists across the tested moduli. The unembedding ratio is the within-block positive-change ratio, not an additive full-kernel attribution. The $p = 59$ row uses a 15,000-epoch budget, $p = 83$ uses 8,500 epochs, $p = 89$ uses 10,000 epochs, $p = 101$ uses 7,500 epochs, $p = 113$ uses the expanded 7,000-epoch baseline, and $p = 127$ uses 5,500 epochs.

Setting	seeds	grokked	med. t_{grok}	unemb. ratio	Spearman $\rho(p)$
$p = 59$	20	10	13425	69.0%	0.868 (6.94×10^{-7})
$p = 83$	20	10	6700	72.0%	0.902 (5.61×10^{-8})
$p = 89$	20	12	8100	68.5%	0.926 (4.94×10^{-9})
$p = 101$	20	12	6150	72.4%	0.802 (2.14×10^{-5})
$p = 113$	20	10	4900	95.8%	0.928 (3.76×10^{-9})
$p = 127$	20	9	4550	99.4%	0.892 (1.31×10^{-7})

Table 18 continued

p	seed	test	t_{grok}	embed	attn	mlp	unembed	grok
83	9	0.281	-	-0.012	+0.004	-0.014	+0.372	N
	10	0.987	14050	-0.013	+0.012	-0.013	+0.584	Y
	11	0.429	-	-0.011	-0.002	-0.013	+0.270	N
	12	0.233	-	-0.008	-0.006	-0.016	+0.312	N
	13	0.974	14500	-0.014	-0.005	-0.010	+0.560	Y
	14	0.993	10850	-0.015	+0.085	-0.010	+0.686	Y
	15	0.999	13750	-0.014	+0.003	+0.140	+0.601	Y
	16	0.998	12800	-0.016	+0.002	-0.015	+0.605	Y
	17	0.256	-	-0.012	-0.007	-0.014	+0.276	N
	18	0.976	14000	-0.014	+0.014	-0.013	+0.562	Y
	19	0.920	-	-0.015	-0.007	+0.015	+0.470	N
	0	0.347	-	-0.008	-0.003	-0.010	+0.242	N
	1	1.000	7700	-0.011	+0.016	+0.120	+0.549	Y
	2	0.138	-	-0.008	-0.002	-0.008	+0.260	N
	3	0.731	-	-0.010	-0.001	-0.011	+0.458	N
	4	1.000	6500	-0.009	+0.001	-0.011	+0.773	Y
	5	0.095	-	-0.008	-0.005	-0.011	+0.216	N
	6	0.649	-	-0.009	+0.001	+0.001	+0.434	N

Continued in next column

Table 16: **Final per-block $\bar{A}^{\text{add}}$ summary by budget outcome class.** Median [min, max] over seeds, last-4-checkpoint means, reported with one row per modulus/outcome split. “Not grokked by budget” rows are right-censored at that modulus’s epoch budget.

p	split	test	embed	attn	mlp	unembed
59		1.00	-0.014	+0.029	-0.010	+0.603
	grokked by budget ($n = 10$)	[0.974, 0.999]	[-0.016, -0.013]	[-0.005, +0.160]	[-0.015, +0.153]	[+0.484, +0.686]
	not grokked by budget ($n = 10$)	0.245	-0.012	-0.006	-0.014	+0.273
83		1.00	-0.011	+0.034	-0.008	+0.719
	grokked by budget ($n = 10$)	[0.955, 1.000]	[-0.011, -0.009]	[-0.003, +0.084]	[-0.011, +0.152]	[+0.549, +0.779]
	not grokked by budget ($n = 10$)	0.414	-0.008	-0.004	-0.010	+0.251
89		0.999	-0.010	+0.035	+0.040	+0.642
	grokked by budget ($n = 12$)	[0.988, 1.000]	[-0.011, -0.009]	[-0.003, +0.125]	[-0.010, +0.157]	[+0.532, +0.761]
	not grokked by budget ($n = 8$)	0.609	-0.008	-0.005	-0.010	+0.394
101		0.999	-0.009	+0.027	+0.001	+0.686
	grokked by budget ($n = 12$)	[0.988, 1.000]	[-0.009, -0.008]	[-0.005, +0.100]	[-0.009, +0.140]	[+0.498, +0.740]
	not grokked by budget ($n = 8$)	0.522	-0.007	-0.004	-0.009	+0.311
113		1.00	-0.008	-0.001	-0.007	+0.645
	grokked by budget ($n = 10$)	[0.994, 1.000]	[-0.009, -0.006]	[-0.006, +0.143]	[-0.008, +0.106]	[+0.462, +0.720]
	not grokked by budget ($n = 10$)	0.179	-0.006	-0.004	-0.008	+0.151
127		1.00	-0.007	-0.006	-0.006	+0.532
	grokked by budget ($n = 9$)	[0.994, 1.000]	[-0.007, -0.007]	[-0.007, +0.130]	[-0.007, +0.118]	[+0.468, +0.748]
	not grokked by budget ($n = 11$)	0.513	-0.006	-0.005	-0.007	+0.286

Table 18 continued

p	seed	test	t_{grok}	embed	attn	mlp	unembed	grok
7	1.000	5050	-0.011	+0.084	+0.090	+0.779	Y	
8	0.833	-	-0.010	-0.004	-0.009	+0.425	N	
9	1.000	5650	-0.011	-0.002	-0.010	+0.754	Y	
10	0.191	-	-0.007	-0.004	-0.011	+0.081	N	
11	0.619	-	-0.011	-0.005	+0.150	+0.426	N	
12	0.218	-	-0.008	-0.005	-0.010	+0.230	N	
13	1.000	6800	-0.010	+0.070	-0.010	+0.750	Y	
14	0.481	-	-0.006	-0.005	-0.011	+0.214	N	
15	0.955	7450	-0.011	+0.001	-0.008	+0.688	Y	
16	0.999	7500	-0.011	+0.072	-0.011	+0.632	Y	
17	1.000	5450	-0.011	+0.054	+0.114	+0.755	Y	
18	0.996	8250	-0.010	-0.003	-0.009	+0.587	Y	
19	0.999	6600	-0.011	+0.052	+0.152	+0.598	Y	
89	0	1.000	3700	-0.009	+0.043	+0.067	+0.736	Y
1	0.051	-	-0.002	-0.002	-0.010	+0.010	N	
2	0.832	-	-0.010	-0.006	-0.010	+0.490	N	
3	0.988	9550	-0.010	+0.005	-0.010	+0.612	Y	
4	0.936	-	-0.010	-0.005	-0.007	+0.498	N	
5	0.998	9150	-0.011	-0.001	+0.014	+0.545	Y	
6	0.638	-	-0.006	-0.003	-0.010	+0.384	N	
7	0.997	9050	-0.011	+0.035	+0.107	+0.556	Y	
8	0.581	-	-0.009	-0.006	+0.014	+0.403	N	
9	0.955	-	-0.008	-0.001	-0.009	+0.464	N	
10	1.000	5600	-0.011	+0.125	+0.157	+0.761	Y	
11	1.000	7100	-0.010	+0.022	+0.001	+0.660	Y	
12	1.000	7350	-0.011	+0.055	+0.146	+0.717	Y	
13	0.110	-	-0.008	-0.006	-0.010	+0.140	N	
14	0.997	8400	-0.010	+0.029	-0.010	+0.652	Y	
15	0.999	7800	-0.011	+0.076	+0.147	+0.631	Y	
16	0.995	9300	-0.010	-0.003	-0.009	+0.532	Y	
17	0.503	-	-0.008	-0.007	-0.008	+0.300	N	
18	1.000	7150	-0.010	+0.115	+0.112	+0.655	Y	
19	1.000	8450	-0.011	+0.034	-0.010	+0.585	Y	
101	0	0.949	-	-0.008	-0.004	-0.009	+0.460	N
1	0.999	7000	-0.009	-0.005	+0.007	+0.498	Y	
2	1.000	3550	-0.009	+0.100	+0.116	+0.717	Y	
3	0.887	-	-0.008	-0.007	-0.010	+0.366	N	
4	0.998	5500	-0.009	+0.034	-0.009	+0.680	Y	
5	1.000	6300	-0.008	+0.005	-0.008	+0.674	Y	
6	0.816	-	-0.009	-0.005	+0.025	+0.480	N	

Continued in next column

Table 17: **Statistical summary by modulus.** MW = Mann–Whitney U landmark comparison, grokked by budget vs. not grokked by budget. Spearman ρ is rank correlation with final test accuracy across all seeds for that modulus.

p	Metric	GB med	NGB med	MW p	Spearman ρ (p)
59	embed	−0.014	−0.012	0.0013	−0.821 (9.21×10^{-6})
	attn	+0.029	−0.006	0.0010	+0.802 (2.10×10^{-5})
	mlp	−0.010	−0.014	0.0211	+0.723 (0.0003)
	unembed	+0.603	+0.273	0.0002	+0.868 (6.94×10^{-7})
83	embed	−0.011	−0.008	0.0058	−0.703 (0.0005)
	attn	+0.034	−0.004	0.0017	+0.764 (8.94×10^{-5})
	mlp	−0.008	−0.010	0.1859	+0.362 (0.1170)
	unembed	+0.719	+0.251	0.0002	+0.902 (5.61×10^{-8})
89	embed	−0.010	−0.008	0.0002	−0.685 (0.0009)
	attn	+0.035	−0.005	6.35×10^{-5}	+0.858 (1.35×10^{-6})
	mlp	+0.040	−0.010	0.0473	+0.573 (0.0083)
	unembed	+0.642	+0.394	1.59×10^{-5}	+0.926 (4.94×10^{-9})
101	embed	−0.009	−0.007	0.0002	−0.692 (0.0007)
	attn	+0.027	−0.004	0.0022	+0.666 (0.0013)
	mlp	+0.001	−0.009	0.0574	+0.574 (0.0081)
	unembed	+0.686	+0.311	1.59×10^{-5}	+0.802 (2.14×10^{-5})
113	embed	−0.008	−0.006	0.0091	−0.669 (0.0013)
	attn	−0.001	−0.004	0.1405	+0.364 (0.1144)
	mlp	−0.007	−0.008	0.4727	+0.479 (0.0325)
	unembed	+0.645	+0.151	0.0002	+0.928 (3.76×10^{-9})
127	embed	−0.007	−0.006	0.0575	−0.624 (0.0033)
	attn	−0.006	−0.005	0.8792	+0.161 (0.4982)
	mlp	−0.006	−0.007	0.0276	+0.559 (0.0103)
	unembed	+0.532	+0.286	0.0008	+0.892 (1.31×10^{-7})

Table 18 continued

p	seed	test	t_{grok}	embed	attn	mlp	unembed	grok	
	7	0.119	–	–0.006	–0.005	–0.008	+0.155	N	
	8	0.298	–	–0.006	–0.004	–0.009	+0.170	N	
	9	0.049	–	–0.003	–0.003	–0.009	+0.034	N	
	10	0.999	4400	–0.009	+0.066	+0.073	+0.740	Y	
	11	1.000	6100	–0.009	+0.012	+0.140	+0.616	Y	
	12	0.077	–	–0.007	–0.005	+0.002	+0.257	N	
	13	0.999	5850	–0.008	+0.020	–0.008	+0.740	Y	
	14	1.000	6300	–0.009	+0.060	+0.102	+0.590	Y	
	15	0.998	6700	–0.009	–0.004	–0.009	+0.533	Y	
	16	0.747	–	–0.007	–0.004	–0.009	+0.388	N	
	17	1.000	6400	–0.009	+0.071	+0.033	+0.730	Y	
	18	0.988	5350	–0.009	+0.009	–0.008	+0.692	Y	
	19	1.000	6200	–0.009	+0.039	–0.004	+0.708	Y	
	113	0	1.000	4800	–0.008	+0.068	–0.008	+0.641	Y
		1	1.000	5650	–0.008	+0.001	–0.008	+0.710	Y
		2	0.290	–	–0.006	–0.004	–0.006	+0.331	N
		3	0.995	6650	–0.007	–0.005	–0.008	+0.462	Y
		4	0.050	–	–0.004	–0.004	–0.008	–0.001	N
		5	0.297	–	–0.008	–0.007	+0.093	+0.446	N
6		0.117	–	–0.007	–0.002	–0.008	+0.166	N	
7		0.084	–	–0.006	–0.005	–0.008	+0.051	N	
8		1.000	3250	–0.006	–0.003	–0.006	+0.659	Y	
9	1.000	4950	–0.008	+0.127	+0.106	+0.720	Y		

Continued in next column

Table 18 continued

p	seed	test	t_{grok}	embed	attn	mlp	unembed	grok
	10	0.211	–	–0.004	–0.003	–0.008	+0.137	N
	11	0.994	6700	–0.008	–0.003	–0.008	+0.485	Y
	12	0.074	–	–0.005	–0.005	–0.008	+0.028	N
	13	1.000	4550	–0.009	–0.006	–0.007	+0.634	Y
	14	1.000	4850	–0.008	+0.143	+0.097	+0.642	Y
	15	0.293	–	–0.007	–0.004	–0.005	+0.339	N
	16	0.929	–	–0.008	–0.005	–0.005	+0.383	N
	17	0.998	6150	–0.008	–0.006	–0.008	+0.649	Y
	18	0.147	–	–0.004	–0.002	–0.008	+0.068	N
	19	1.000	3150	–0.006	+0.004	–0.006	+0.679	Y
127	0	0.955	–	–0.006	–0.005	–0.007	+0.409	N
	1	1.000	2600	–0.007	–0.003	–0.007	+0.748	Y
	2	0.902	–	–0.007	–0.006	–0.007	+0.381	N
	3	0.128	–	–0.006	–0.005	–0.007	+0.188	N
	4	0.127	–	–0.004	–0.004	–0.007	+0.132	N
	5	1.000	4700	–0.007	–0.006	–0.006	+0.523	Y
	6	0.578	–	–0.006	–0.005	–0.006	+0.286	N
	7	1.000	3100	–0.007	–0.007	–0.007	+0.651	Y
	8	0.961	–	–0.007	–0.005	–0.007	+0.435	N
	9	0.998	4800	–0.007	–0.006	–0.007	+0.522	Y
	10	1.000	2850	–0.007	+0.130	+0.118	+0.641	Y
	11	0.513	–	–0.007	–0.006	–0.007	+0.209	N
	12	0.472	–	–0.007	–0.004	–0.006	+0.338	N
	13	0.994	5100	–0.007	–0.006	+0.062	+0.476	Y
	14	1.000	4550	–0.007	–0.004	–0.007	+0.680	Y
	15	1.000	5100	–0.007	–0.006	–0.006	+0.468	Y
	16	0.035	–	–0.006	–0.006	–0.008	–0.003	N
	17	1.000	4400	–0.007	+0.068	+0.115	+0.532	Y
	18	0.163	–	–0.006	–0.005	–0.007	+0.076	N
19	0.846	–	–0.007	–0.005	+0.037	+0.571	N	

K Conditional Epoch-Scaling Calculation

This appendix presents a proof-style scaling calculation motivated by the norm-separation account of grokking delay [Khanh et al., 2026]. The statement is intentionally conditional. It clarifies why, under a fixed-minibatch convention, the number of *epochs* between memorization and grokking may decrease as the modulus increases, even if the underlying optimizer-step delay varies only slowly. This is not a universal monotonicity law in p , and it should not be confused with our full-batch eNTK runs, where one epoch corresponds to one optimizer update.

Let $\Delta_{\text{step}}(p) = T_{\text{grok}}(p) - T_{\text{mem}}(p)$ represent the delay measured in optimizer steps, and let $\Delta_{\text{epoch}}(p)$ represent the same delay measured in epochs. Let $\theta_{\text{mem}}(p)$ be a memorizing interpolant with high norm, $\theta_{\text{post}}(p)$ a post-grokking solution with lower norm, and $\gamma_{\text{eff}}(p)$ the effective contraction rate of the optimizer.

Assumption 1 (Norm-separation delay law). *For the modulus range under consideration, the expected step delay obeys*

$$\mathbb{E}[\Delta_{\text{step}}(p)] \leq C\gamma_{\text{eff}}(p)^{-1}\mathbb{E}\left[\log \frac{\|\theta_{\text{mem}}(p)\|^2}{\|\theta_{\text{post}}(p)\|^2}\right] + c_0, \quad (10)$$

for constants $C, c_0 > 0$ independent of p across the experimental range.

Assumption 2 (Finite-width norm compression). *The expected logarithmic norm ratio*

$$L(p) := \mathbb{E}\left[\log \frac{\|\theta_{\text{mem}}(p)\|^2}{\|\theta_{\text{post}}(p)\|^2}\right] \quad (11)$$

is nonincreasing over the modulus range of interest.

Assumption 3 (No adverse optimizer slowdown). *The effective contraction rate is bounded below: $\gamma_{\text{eff}}(p) \geq \gamma_0 > 0$ over the same modulus range.*

Proposition 16 (When expected epoch delay can decrease with larger modulus). *Consider modular addition over $\mathbb{Z}_p$, with training fraction $\rho \in (0, 1]$ and fixed minibatch size B . Under Assumptions 1–3, the expected epoch delay satisfies*

$$\mathbb{E}[\Delta_{\text{epoch}}(p)] \lesssim \frac{B}{\rho p^2} \left(\frac{C}{\gamma_0} L(p) + c_0 \right), \quad (12)$$

up to minibatch-rounding effects. Consequently, when $L(p)$ is nonincreasing, this upper bound decreases at least by the explicit p^{-2} conversion factor, apart from the additive floor and ordinary finite-sample effects.

Proof. Assumption 1 gives the expected optimizer-step delay. Since $\gamma_{\text{eff}}(p) \geq \gamma_0$,

$$\mathbb{E}[\Delta_{\text{step}}(p)] \leq \frac{C}{\gamma_0} L(p) + c_0. \quad (13)$$

At fixed training fraction ρ and fixed minibatch size B , the modular-addition training set has approximately ρp^2 examples, so the number of minibatch updates per epoch is

$$S(p) \approx \frac{\rho p^2}{B}. \quad (14)$$

Converting optimizer steps to epochs gives

$$\mathbb{E}[\Delta_{\text{epoch}}(p)] \approx \frac{\mathbb{E}[\Delta_{\text{step}}(p)]}{S(p)} \leq \frac{B}{\rho p^2} \left(\frac{C}{\gamma_0} L(p) + c_0 \right), \quad (15)$$

which proves Eq. 12. If $L(p)$ is nonincreasing, then the numerator is nonincreasing while p^2 is increasing, so the bound decreases with p up to discretization and finite-sample effects. $\square$

Two factors drive the calculation. The norm-separation framework predicts shorter step delays when increasing p reduces the norm ratio between memorizing and post-grokking solutions. Separately, with a fixed minibatch size, each epoch involves $\Theta(p^2)$ optimizer updates because the full modular-addition table has p^2 input pairs. Even a constant step delay would thus correspond to an $O(p^{-2})$ epoch delay; a decreasing step delay only amplifies this effect.

The roughly half-grokged-by-budget/half-not-grokged-by-budget division is controlled by a seed-wise grok-time quantile rather than directly by an expectation. If $T_{\text{grok}}^{(s)}(p)$ denotes seed s 's grokking time, a budget yielding an approximately 50/50 landmark split lies close to the median $q_{0.5}(p)$. Proposition 16 bounds the expected post-memorization delay; the median budget $q_{0.5}(p)$ decreases with p provided that memorization time and seed-to-seed variability do not increase enough to counter that bound. This is the specific way in which the calculation can explain decreasing epoch budgets for a half-split; runs on the not-grokged-by-budget side remain right-censored. In our reported eNTK experiments the training loop is full-batch, so the p^{-2} minibatch-conversion factor does not apply directly. The observed modulus-specific budgets are empirical selections, while the norm-separation argument provides a conditional justification for decreasing budgets in related fixed-minibatch regimes.

This calculation serves as a safeguard for interpreting modulus sweeps rather than as primary evidence for unembedding localization. The main claim is blockwise and representation-geometric: following independent centering and trace-normalization, the unembedding-block eNTK precisely corresponds to the residual-stream Gram matrix and empirically contains the addition-aligned signal.

L Run Configuration

The completed modulus sweep uses the run configurations in Table 19.

The budgets in Fig. 5 are empirical median-targeting choices rather than a strictly monotone fitted schedule. For each modulus we used 20 seeds and selected an epoch budget intended to leave roughly half the seeds grokged by budget and half not grokged by budget, i.e. about 10 grokged-by-budget seeds. With only 20 seeds, the seed-wise median grok time is noisy, and nearby moduli can therefore show small local inversions. The $p = 83$ row uses fewer epochs than the $p = 89$ row (8,500 vs. 10,000) for this finite-seed reason; the aggregate sweep still shows the intended decreasing-budget trend from $p = 59$ to $p = 127$.

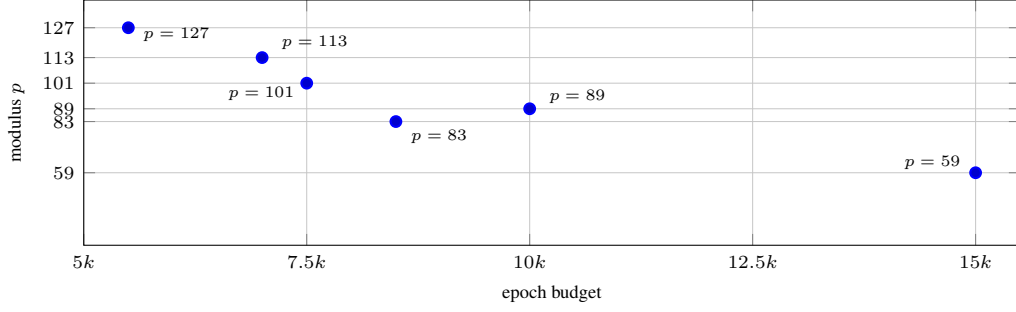


Figure 5: **Epoch budgets for completed modulus sweeps.** Points mark the researcher-chosen training budgets used for each modulus to obtain informative landmark splits. They are design choices rather than random outcome observations, so no inferential regression band is shown.

Table 19: **Per-modulus run configurations.** Completed rows correspond to experiments reported in this paper. All rows use train fraction $\rho = 0.30$, the baseline architecture, AdamW with lr 10^{-3} and weight decay 1.0, and 6 complete sum-orbits as the probe set unless otherwise noted.

p	seeds	epochs	probe orbits	probe $n = 6p$	eNTK checkpoints
59	20	15000	6	354	every 500 epochs
83	20	8500	6	498	every 500 epochs
89	20	10000	6	534	every 500 epochs
101	20	7500	6	606	every 500 epochs
113	20	7000	6	678	every 500 epochs
127	20	5500	6	762	every 500 epochs

Field	Value (completed baseline / per-modulus expansion)
Modulus p	completed: 59, 83, 89, 101, 113, 127
Train fraction ρ	0.30
Weight decay	1.0
Seeds	20 completed for $p \in \{59, 83, 89, 101, 113, 127\}$
Transformer	1 layer, 4 heads, $d_{\text{model}} = 128$, $d_{\text{mlp}} = 512$, no LayerNorm
Total parameters	$\approx 227,000$
Block sizes	embed 6.6%, attn 29.1%, mlp 57.9%, unembed 6.4%
Optimizer	AdamW (lr 10^{-3} , $\beta = (0.9, 0.98)$)
Epochs	15,000 for $p = 59$; 8,500 for $p = 83$; 10,000 for $p = 89$; 7,500 for $p = 101$; 7,000 for $p = 113$; 5,500 for $p = 127$
Probe set	6 orbits, $n = 6p$ for every modulus
$\text{rank}(P_{\text{add}})$	6 (incl. constant mode)
Centered informative modes	5
m for $\bar{A}^{\text{add}}$	6 main; 5 sensitivity
eNTK checkpoints	every 500 epochs (completed: 30 total for $p = 59$, 17 for $p = 83$, 20 for $p = 89$, 15 for $p = 101$, 14 for $p = 113$, 11 for $p = 127$)
eNTK estimator	$R = 16$ Rademacher projections (with $R \in \{32, 64\}$ robustness check)
Accuracy evaluation	every 50 epochs
t_{grok} threshold	first evaluation in two consecutive evaluations with test acc $> 95\%$
Interventions	freeze W_U pre-grok; reinit W_U post-grok; matched-size embedding-block freeze (W_E)

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [\[Yes\]](#)

Justification: The abstract, Introduction, Discussion, and Conclusion distinguish normalized shape alignment, additive attribution, and intervention-based functional relevance, explicitly restricting necessity and partial-sufficiency claims to the ten evaluated grokked primary models.

Guidelines:

- The answer [\[N/A\]](#) means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [\[No\]](#) or [\[N/A\]](#) answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: The dedicated Limitations section discusses censoring, architecture and task scope, pathway measurements, statistical limitations, and intervention limitations.

Guidelines:

- The answer [\[N/A\]](#) means that the paper has no limitation while the answer [\[No\]](#) means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate “Limitations” section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren’t acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The estimator proposition is proved in the main text, which also derives the readout identity, its Fourier interpretation, and the upstream gradient-kernel factorization with their assumptions and scope.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The Setup and artifact specify the task construction, architecture, optimizer, data split, model and estimator seeds, checkpoints, probe sets, metrics, landmark definitions, bootstrap seeds, and run configurations. The causal controls additionally release all 1,640 draw-level rows and the deterministic random-basis seed rule.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

- (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The active archive at <https://anonymous.4open.science/r/entk-grokking-5F13> includes runnable primary and locked-replication implementations, machine-readable protocols, deterministic seeds, all reported per-seed result tables, draw-level causal controls, statistical and plotting scripts, reproduction commands, and checksums. The released modulus outputs reproduce the reported cross-modulus analyses; limitations on full modulus retraining, bitwise environment reconstruction, and auxiliary figure writers are disclosed in the appendix.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://neurips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The Setup reports the data split, architecture, optimizer, learning rate, weight decay, epoch budgets, checkpoint schedule, probe construction, estimator settings, and grokking criterion.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The main claims are supported by Mann–Whitney, Spearman, and paired Wilcoxon tests across independently initialized seeds, with reported p -values, seed counts, bootstrap confidence intervals, and exact per-seed values in the artifact.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The paper reports one GPU per process; NVIDIA RTX 6000 Pro, H100, and A100 reference hardware; reconstructed training-only and eNTK-only timings with plausible ranges; estimated peak allocated and reserved memory; combined per-seed costs; and approximate archived (1.4 GPU-hours) and full-project (2.5 GPU-hours) totals including exploratory, failed, and repeated runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The work uses synthetic arithmetic data, preserves submission anonymity, involves no human participants or sensitive data, and reports limitations and negative findings relevant to responsible scientific interpretation.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The Broader impacts paragraph in the Limitations section discusses the methodological benefit and the principal risk of overinterpreting diagnostic alignment as causal localization.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The work releases no high-capability model, scraped dataset, or other asset presenting a material misuse risk; it studies small models trained on synthetic modular arithmetic.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [N/A]

Justification: The experiments use procedurally generated modular-arithmetic data and do not rely on an existing dataset, pretrained model, or redistributed third-party asset. Related methods and prior work are cited conventionally.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The submission does not currently introduce a separately released dataset, model, or code asset.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The research involves neither crowdsourcing nor human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The research involves no human participants, personal data, or participant-facing procedures, so IRB review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not an important, original, or non-standard component of the scientific methods or experiments reported in the paper.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.