Mechanistic Interpretability of Grokking in MLPs: Tracking Fourier Feature Emergence Across Modular Arithmetic Operations

Paul Dubrulle · ISYE 4803: Foundations of Modern Data Science

I. Introduction

Grokking is a phenomenon discovered by Power et al. [1] where, on algorithmic tasks, neural networks (NN) initially memorize the training data and then, after an extended period of training, suddenly generalize. This decoupling of memorization from generalization that gives the ability to study them as distinct phases rather than as a single blurred event is why its interesting. Mechanistic Interpretability (MI) is an approach to understanding the inner workings of neural networks by examining the learned weights, activations, and circuits to discover structures or algorithms hidden within. Understanding how NNs learn to generalize and exploring the MI approach, from which we are able to extract a notion of progress, are both a step towards more intelligible artificial intelligence (AI).

Nanda et al. [2] applied MI to a grokked transformer trained on modular addition, discovering interpretable Fourier features within its hidden layers. They defined progress measures from this mechanistic understanding, identifying three discrete training phases (memorization, circuit formation, cleanup) and showing that grokking is not actually a sudden shift but rather slow continuous hidden progress before test accuracy jumps. Where Nanda et al. characterize the *trajectory* of grokking, Gromov [3] characterizes its *endpoint*: he derived an analytical closed-form solution for the weights of a 2-layer MLP that solves modular addition, showing the weight rows become sinusoidal at specific frequencies k/p. Doshi et al. [4] extended this to subtraction, multiplication, and division, with the key insight that the multiplication and division solutions live in the discrete logarithm basis.

The trajectory of how MLPs reach the closed-form solutions of Gromov and Doshi remains uncharacterized; Nanda et al. [2] traced this trajectory only for a transformer on addition. This leaves several open questions: Which Fourier modes appear first during training, and how does the spectrum sharpen toward the analytical solution? Do the four operations follow comparable or qualitatively different trajectories, particularly between additive and multiplicative tasks? And for multiplication and division, does the discrete logarithm basis structure emerge simultaneously with, before, or after Fourier concentration? Answering these questions in the MLP setting bridges the endpoint and trajectory views of grokking and tests whether Nanda et al.'s findings extend beyond the transformer architecture and the addition operation.

We address these questions by training identical 2-layer ReLU MLPs on each of the four operations across 5 random seeds and tracking Fourier emergence using the Inverse Participation Ratio (IPR) of the learned weights. Hidden progress extends from transformers on addition to MLPs on all four operations, and the IPR trajectories collapse onto a single universal curve once multiplication and division are evaluated in the discrete-log basis of Doshi et al. [4]. For the multiplicative tasks, log-basis structure and Fourier concentration emerge simultaneously rather than sequentially, refuting a natural two-stage hypothesis. The endpoint weights match the closed-form predictions of Gromov [3] and Doshi et al. [4] across all four operations, showing that the trajectory and endpoint views of grokking describe the same underlying solution.

II. Background and Related Work

A. Grokking and Delayed Generalization

While grokking has been observed across architectures and operations, its occurrence and timing are sensitive to hyperparameters and dataset size. Power et al. [1] showed that grokking time depends strongly on training data fraction, with smaller fractions producing longer delays and a critical threshold below which generalization fails entirely, and that explicit regularization such as weight decay is generally required for grokking to occur. Liu et al. [5] formally distinguish four learning phases through phase diagrams over learning rate and weight decay: comprehension, grokking, memorization, and confusion. Comprehension, the desired training regime in practice, shows train and test improving together, while grokking specifically requires train convergence well before test convergence, and is characterized as an outcome of improperly tuned hyperparameters sandwiched between comprehension and memorization. Davies et al. [6] further argue that grokking and the classical double descent phenomenon are instances of the same underlying competition between fast- and slow-to-learn patterns. This phase structure makes deliberately landing in the grokking regime a methodological consideration we return to in Section III.

B. Mechanistic Interpretability of Grokking

Nanda et al. [2] reverse-engineered the algorithm a 1-layer transformer learns when grokking modular addition: each integer is embedded onto rotations $\{\cos(w_k a), \sin(w_k a)\}$ at a sparse set of frequencies w_k, and the MLP combines pairs through trigonometric identities such as $\cos(w_k a)\cos(w_k b) - \sin(w_k a)\sin(w_k b) = \cos(w_k(a+b))$ to produce the modular sum. To track this circuit's emergence, they introduced two progress measures based on the network's logits (its raw, pre-softmax output values): restricted loss, computed by keeping only the key Fourier frequencies in the logits, and excluded loss, computed by removing only those frequencies. They

additionally use a Gini coefficient (a scalar measure of distributional concentration, where higher values indicate greater sparsity) applied to the Fourier components of the embedding and unembedding matrices. These measures revealed that the Fourier circuit forms gradually during the apparent memorization plateau, well before the test-loss phase transition. Our work adopts this framework in the MLP setting, with the Inverse Participation Ratio (IPR) and the raw Fourier spectrum of the weights as our specific progress measures.

C. Analytical Solutions for Modular Arithmetic in MLPs

Gromov [3] derived a closed-form solution for the weights of a 2-layer MLP that solves modular addition. The architecture takes as input the concatenation of two one-hot vectors $\mathbf{e}_a, \mathbf{e}_b \in \mathbb{R}^p$ for inputs $a, b \in \{0, 1, \ldots, p-1\}$, passes through a hidden layer of width N, and outputs p logits indexed by q. For the task $f(a, b) = a + b \bmod p$, the optimal first-layer weights factor into two $N \times p$ blocks corresponding to the two inputs, and both layers take the form

$$W^{(1)}_{kn} = \begin{pmatrix} \cos\left(\frac{2\pi k}{p} n_1 + \varphi^{(1)}_k\right) \\ \cos\left(\frac{2\pi k}{p} n_2 + \varphi^{(2)}_k\right) \end{pmatrix}, \qquad W^{(2)}_{qk} = \cos\left(-\frac{2\pi k}{p} q - \varphi^{(3)}_k\right), \tag{1}$$

with the phase constraint $\varphi^{(1)}_k + \varphi^{(2)}_k = \varphi^{(3)}_k$. When summed over k, all spurious cross-terms have random phases and cancel by destructive interference, leaving only the term that fires when $a + b \equiv q \pmod p$, effectively a Fourier-basis representation of the modular delta function.

Subtraction reduces to addition through the additive inverse, so its analytical solution is Gromov's with a sign-flipped index. Multiplication and division require a different treatment because the multiplicative structure of integers mod p has period $p - 1$ rather than p: for any prime p, there exists a primitive element g such that the nonzero residues $\{1, 2, \ldots, p-1\}$ can be enumerated as $\{g^0, g^1, \ldots, g^{p-2}\}$. The discrete logarithm $\log_g(r)$, defined as the exponent $r' \in \{0, \ldots, p-2\}$ such that $g^{r'} = r$, converts multiplication into addition:

$$\log_g(a \cdot b) \equiv \log_g a + \log_g b \pmod{p-1}. \tag{2}$$

Doshi et al. [4] use this identity to extend Gromov's framework: the closed-form solution for multiplication is structurally identical to addition's, with weight rows sinusoidal at frequencies $k/(p-1)$ when the column indices are relabeled by $\log_g$. The element 0 has no logarithm and is handled separately, which also means $b = 0$ is excluded for division, yielding $p(p-1)$ valid pairs rather than p^2. These analytical solutions provide the targets against which our progress measures are evaluated: a fully grokked MLP should display weight rows sinusoidal at frequencies k/p for additive operations and at frequencies $k/(p-1)$ in the $\log_g$ basis for multiplicative operations.

III. Methodology

We train identical 2-layer ReLU MLPs on each of the four modular operations and track the emergence of the Fourier structure predicted by (1) using progress measures defined directly on the learned weights. The architecture and dataset construction match those of [3], [4] so that our trajectory measurements speak to their endpoint analyses, and the training protocol is chosen to land squarely in the grokking phase of [5]. Throughout this section, $\star \in \{+, -, \cdot, /\}$ denotes the operation under study.

A. Task and Architecture

We use the 2-layer MLP analyzed in §II-C with $p = 113$ (matching [2]), hidden width $N = 128$, and no additive biases, so that trained weights are directly comparable to (1). For analysis we treat $W^{(1)}$ as two $N \times p$ blocks $W^{(1)}_a$ and $W^{(1)}_b$ corresponding to the two inputs.

For addition and subtraction, the dataset consists of all $p^2 = 12{,}769$ pairs $(a, b) \in \{0, \ldots, p-1\}^2$ with one-hot target $\mathbf{e}_y$ where $y = (a \star b) \bmod p$. For multiplication and division we exclude $b = 0$, yielding $p(p-1) = 12{,}656$ valid pairs; the modular inverse for division is computed via Fermat's little theorem, $b^{-1} \equiv b^{p-2} \pmod p$. Each operation's pairs are shuffled with a per-seed permutation and split with $\rho_{\text{train}} = 0.3$ for training and 0.7 for evaluation. This small training fraction, combined with the regularization choices below, places each run in the grokking phase of [5]'s phase diagram rather than the comprehension or memorization phases.

B. Training Protocol

We optimize the full-batch mean-squared error against the one-hot target with AdamW, learning rate 10^{-3}, and weight decay 1.0. The MSE objective on one-hot targets matches the analytical setup of [3], [4], so the trained weights and the closed-form solution can be compared as solutions to the same optimization problem. Each (operation, seed) run trains for $10{,}000$ steps with seeds $\{0, 1, 2, 3, 4\}$, for a total of 20 runs. Loss, accuracy, and scalar progress measures are logged every 50 steps; full weight snapshots are saved every 200 steps for offline reanalysis. Cross-operation comparisons report the mean across seeds with a ± 1 standard deviation shaded band.

C. Progress Measures

Where [2] use logit-based measures tied to a transformer's residual stream, we use weight-based measures, taking advantage of the fact that in the 2-layer MLP setting the weights themselves are the learned algorithm of (1). For multiplication and division, all Fourier-domain measures below are evaluated in both the natural basis and the discrete-log basis introduced in §II-C; the gap between them is itself a measurement of when the multiplicative structure appears.

a) Inverse Participation Ratio (IPR): For a single weight row $w \in \mathbb{R}^p$ with discrete Fourier transform $\hat{w} = \mathcal{F}[w]$ and normalized power spectrum $\tilde{P}_k = |\hat{w}_k|^2 / \sum_j |\hat{w}_j|^2$, the IPR is

$$\mathrm{IPR}(w) = \sum_{k=0}^{p-1} \tilde{P}_k^2 \in [1/p,\, 1]. \tag{3}$$

A uniform spectrum gives the baseline $1/p$ and a single-mode spike gives 1, so IPR scalarizes how concentrated a row is in Fourier space. We report the per-row IPR averaged across the N hidden neurons, separately for $W_a^{(1)}$, $W_b^{(1)}$, and the transpose $(W^{(2)})^\top$. The transpose is essential: in (1), $W_{qk}^{(2)}$ is sinusoidal in the output index q, not the hidden index k, so the FFT must run along the size-p axis.

b) Per-row Fourier spectrum: Where IPR collapses each row to a scalar, the row-wise normalized power spectrum $\tilde{P}_{n,k}$ shows which frequencies are active. We render it as a $(N \times \lfloor p/2 \rfloor)$ heat map, keeping only the first half of the spectrum (informative for real inputs), with rows sorted by their final-snapshot peak frequency so that emerging structure reads as horizontal banding.

c) Raw weight inspection: For qualitative comparison to (1) we plot $W_a^{(1)}$ and $W_b^{(1)}$ directly with rows sorted by dominant frequency. A fully grokked solution should display visible sinusoidal striping in each row.

d) 2D Fourier of the correct-class logit: Following [2], we form the matrix

$$M[a, b] = f(\mathbf{e}_a, \mathbf{e}_b)_{(a \star b) \bmod p} \tag{4}$$

of the logit assigned to the correct class for every input pair, and take its 2D FFT. A circuit implementing the trigonometric-identity algorithm of [2] produces sparse peaks in $|\mathcal{F}_{2D}[M]|^2$ on the diagonal $\omega_a = \omega_b$ for additive operations, and on the corresponding diagonal in the log basis for multiplicative operations. This measure is operation-agnostic and end-to-end, complementing the layer-wise IPR.

D. Loss Landscape Analysis

To complement the spectrum-based view with a geometric one, we project each weight trajectory $\{(W_t^{(1)}, W_t^{(2)})\}$ to two dimensions via PCA on the flattened, mean-subtracted snapshots; for both add and mul (seed 0), the top two principal directions explain at least 86% of the trajectory variance. We evaluate train and test MSE on a grid in this plane and plot the resulting $\log_{10}$-loss surfaces with the projected trajectory overlaid. Because PCA selects directions the optimizer was happy to traverse, the resulting surface is biased toward looking like a basin; as a sanity check we also evaluate loss along filter-normalized random directions in the sense of [7].

IV. Results

We trained a 2-layer ReLU MLP on each of the four modular operations across 5 random seeds for 20 runs total, and present the results in roughly the order of the questions posed in §I: that grokking is reliably reproduced (§IV-A), that hidden progress generalizes from transformers to MLPs (§IV-B), that the four trajectories are universal under basis correction (§IV-C), that log-basis structure and Fourier concentration emerge simultaneously rather than sequentially (§IV-D), and that the endpoint weights match the closed-form predictions of [3], [4] (§IV-E).

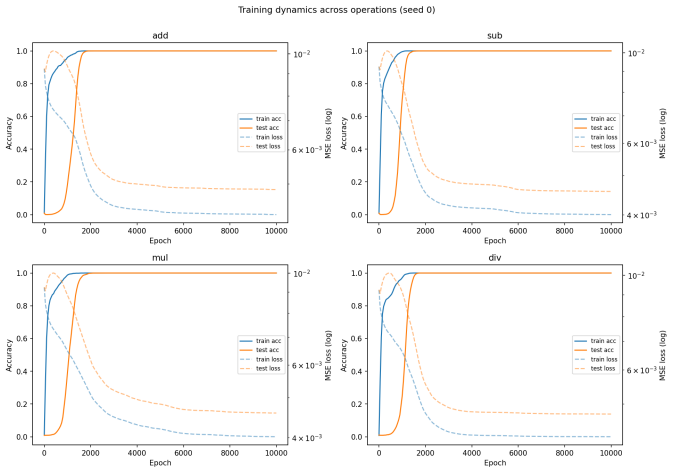

Fig. 1. Train and test accuracy and loss for the four modular operations, mean across 5 seeds with $\pm 1\sigma$ shaded bands. All 20 runs exhibit delayed generalization: train accuracy saturates near epoch 250 while test accuracy remains near chance until approximately epoch 1500.

A. Training Dynamics Across Operations

All 20 runs reach 100% test accuracy with the canonical grokking signature, with train accuracy saturating by approximately epoch 250 and test accuracy following only by approximately epoch 1500 (slightly later for multiplication and division). The train and test loss curves of Fig. 1 show the characteristic test-loss bump near the onset of grokking that distinguishes this regime from the comprehension phase of [5]. This confirms our hyperparameter choices place each run inside the grokking phase rather than the comprehension or memorization phases, providing the long memorization plateau in which the progress measures below are evaluated.

B. Hidden Progress Generalizes to MLPs

For every operation, the IPR of the first-layer weights climbs smoothly from the random baseline $1/p \approx 0.009$ toward approximately 0.4 during the memorization plateau, well before any movement in test accuracy (Fig. 2). The structural circuit predicted by Eq. (1) therefore forms gradually during the apparent plateau in external metrics, generalizing the hidden-progress finding of [2] from a 1-layer transformer on addition to 2-layer MLPs on all four modular operations. The trajectory is not specific to architecture or operation: it reflects a property of the underlying learning dynamics.

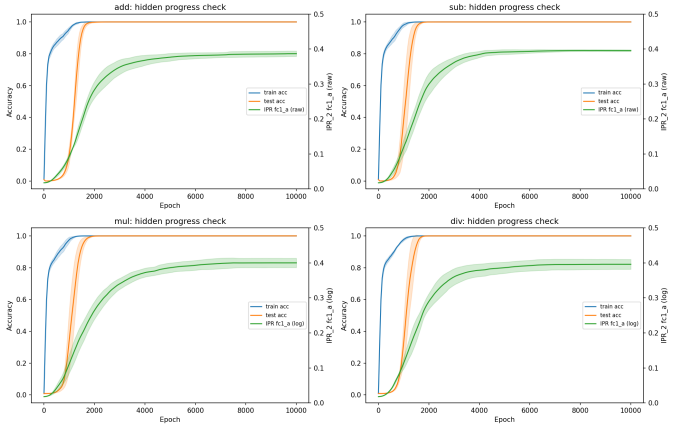

Fig. 2. Hidden progress across operations. Train accuracy (blue) saturates well before test accuracy (orange) rises, and during this gap the IPR of $W_a^{(1)}$ (green, in the correct basis) climbs smoothly from baseline toward ~ 0.4. Mean across 5 seeds with $\pm 1\sigma$ shading.

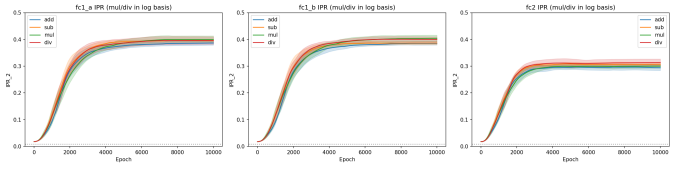

Fig. 3. IPR trajectories across operations for the three weight blocks. All four operations follow the same trajectory within seed variance. Dashed line marks the random baseline $1/p$.

C. A Universal Trajectory Under Basis Correction

With multiplication and division evaluated in the discrete-log basis introduced in §II-C, the IPR trajectories of all four operations overlap within their seed bands across all three weight blocks $W_a^{(1)}$, $W_b^{(1)}$, and $(W^{(2)})^\top$ (Fig. 3). Each block rises from the random baseline beginning around epoch 700 and saturates near 0.4 for the first-layer blocks and near 0.3 for the readout, with the four operations indistinguishable up to seed variance. Once the appropriate basis is fixed, the trajectory is universal: the operation-specific aspect of the learning problem is the basis itself, not the dynamics within it. This bridges the endpoint analyses of [3], [4] with the trajectory analysis of [2] into a single picture.

D. Log-basis Structure and Fourier Concentration Emerge Simultaneously

A natural prediction from the structure of Doshi et al.'s closed form is that learning the multiplicative tasks requires two distinct steps: first acquiring the discrete-log permutation that converts multiplication into addition, then concentrating Fourier energy at the resulting frequencies. Fig. 4 refutes this two-stage hypothesis. The log-basis IPR of $W_a^{(1)}$ (red) rises in unison with test accuracy (gray) for both multiplication and division, while the raw-basis IPR (blue) saturates at approximately 0.08 throughout training and never approaches the log-basis value. The two structures appeared together, we observe no temporal separation between log-basis emergence

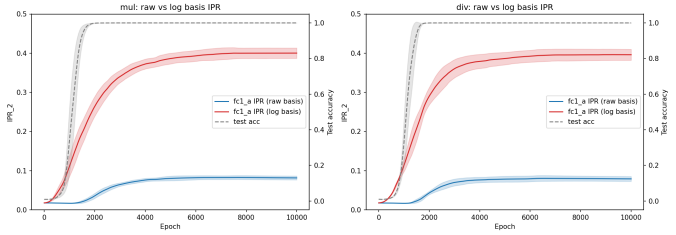

Fig. 4. Testing the two-stage hypothesis. Log-basis IPR (red) of $W_a^{(1)}$ rises together with test accuracy (gray dashed) for both multiplication and division, while raw-basis IPR (blue) saturates at ~ 0.08. The discrete-log basis structure and Fourier concentration emerge simultaneously rather than sequentially.

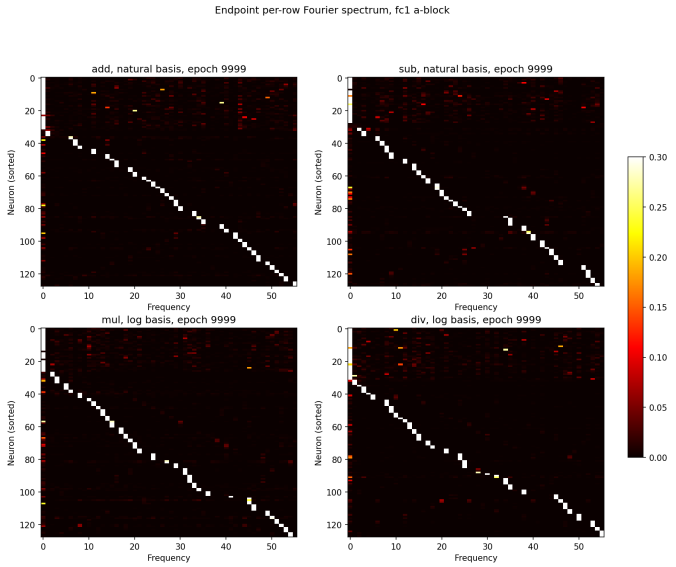

Fig. 5. Fourier Power spectrum of $W_a^{(1)}$ sorted by peak frequency.

and Fourier concentration. The network does not first learn a permutation and then sharpen its features against it but instead reaches a solution in which both properties hold jointly.

E. Endpoint Weights Match the Closed Form

At the end of training, the per-row Fourier spectra of $W^{(1)}$ in the correct basis show approximately 70–80 of the 128 hidden neurons each locked onto a single frequency, producing the diagonal banding predicted by Eq. (1) (Fig. 5). The remaining neurons sit near DC and are effectively decayed under the weight-decay regularization.

The filmstrip in Fig. 6 shows that this banded structure is not present at initialization and emerges progressively during the memorization plateau, with neurons gradually rather than abruptly committing to specific frequencies as training proceeds.

The endpoint IPR values quantify this convergence. In the correct basis, $W_a^{(1)}$ and $W_b^{(1)}$ reach mean IPR values around 0.40 across all four operations with seed standard deviation below 0.015, and $W^{(2)}$ reaches approximately 0.30 (Fig. 7). In the raw basis for multiplication and division, $W^{(1)}$ collapses to approximately 0.08 and $W^{(2)}$ to the random baseline, demon-

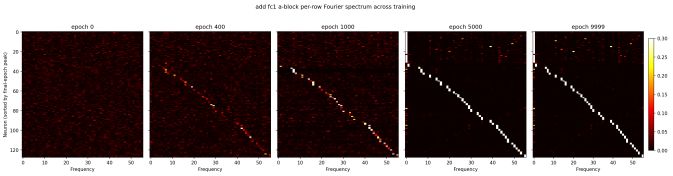

Fig. 6. Evolution of the per-row Fourier spectrum of $W_a^{(1)}$ for modular addition at five training checkpoints. Random structure at initialization gives way to a gradually sharpening diagonal during the memorization plateau, with full crystallization by epoch 5000.

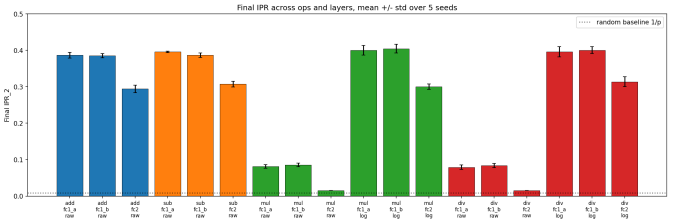

Fig. 7. Endpoint IPR for each operation and weight block, mean $\pm 1\sigma$ over 5 seeds. In the correct basis, $W^{(1)}$ reaches ~ 0.4 and $W^{(2)}$ reaches ~ 0.3 across all four operations. In the raw basis for multiplication and division, $W^{(1)}$ collapses to ~ 0.08 and $W^{(2)}$ to baseline $1/p \approx 0.009$ (dashed line).

strating that the structure is genuinely basis-specific rather than a property of the raw weight values. The intermediate value of 0.08 is itself informative: the raw-basis weights are not pure noise (which would sit at $1/p \approx 0.009$) and are consistent with weights that encode meaningful structure under a permutation of their indices.

The activation-level view confirms the same picture. Plotting the deviation of the correct-class logit from its mean across all input pairs (Fig. 8) reveals the dominant cosine wave predicted by the trigonometric-identity algorithm of [2], with smaller-amplitude residual modes layered on top.

The 2D Fourier transform of the same correct-class logit grid concentrates power on a sparse diagonal $\omega_a = \omega_b$ for the additive operations and on the corresponding diagonal in the log basis for the multiplicative ones (Fig. 9), confirming the trigonometric-identity circuit at the level of the network's outputs.

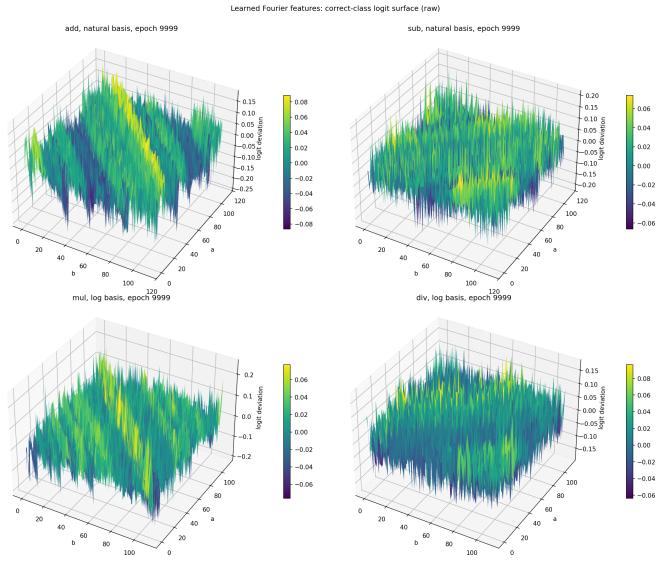

Fig. 8. Deviation of the correct-class logit from its mean as a function of input pair (a, b) at epoch 9999. The dominant cosine wave predicted by the trigonometric-identity circuit of [2] is visible across all four operations, with smaller residual modes overlaid. Multiplication and division are shown in the discrete-log basis.

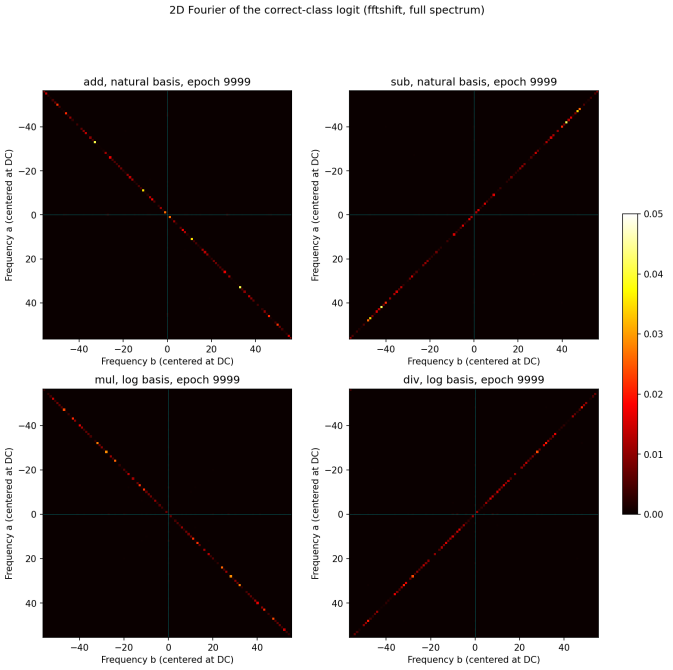

Fig. 9. 2D Fourier transform of the correct-class logit grid at epoch 9999. Energy concentrates on a sparse diagonal $\omega_a = \omega_b$ for additive operations and on the corresponding diagonal in the log basis for multiplicative ones, confirming the trigonometric-identity circuit of [2] at the activation level.

F. Loss Landscape Geometry

To complement the spectrum-based view with a geometric one, we project the weight trajectory of a representative addition run onto the top two principal components of its weight snapshots, capturing 89% of the trajectory variance,

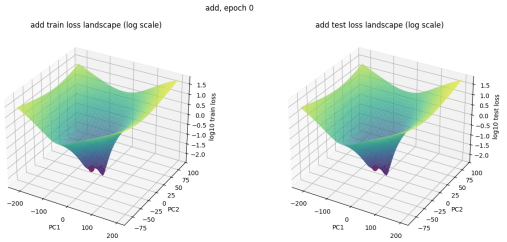

Fig. 10. Train and test loss landscape for modular addition projected onto the top two PCA directions of the weight trajectory (seed 0; 89% variance explained). The grokked solution sits at the bottom of a clean basin in both surfaces.

and evaluate the train and test loss on a grid in this plane (Fig. 10). The grokked solution sits at the bottom of a clean basin in both train and test surfaces with comparable curvature, consistent with a generalizing rather than overfit minimum. The qualitative geometry is similar for the multiplication run on which we replicated this analysis (not shown).

G. Summary of Findings

These results answer the three questions posed in §I. First, the per-row spectra and filmstrip indicate that Fourier modes do not appear in a systematic temporal order; rather, the spectrum sharpens monotonically during the memorization plateau, with neurons gradually committing to specific frequencies as training proceeds. Second, after applying the discrete-log basis correction to multiplication and division, the IPR trajectories of all four operations collapse onto a single curve within seed variance, indicating that the basis is the only operation-specific aspect of the learning problem at the trajectory level. Third, for multiplication and division, the discrete-log basis structure and Fourier concentration emerge simultaneously, refuting the two-stage hypothesis suggested by the structure of the closed-form solution. Together, these findings bridge the endpoint analyses of [3], [4] with the trajectory analysis of [2], extending the latter's hidden-progress framework from a 1-layer transformer on addition to 2-layer MLPs on all four basic modular arithmetic operations.

REFERENCES

[1] A. Power, Y. Burda, H. Edwards, I. Babuschkin, and V. Misra, "Grokking: Generalization beyond overfitting on small algorithmic datasets," in *ICLR Workshop on Mathematical Reasoning in General Artificial Intelligence*, 2022.

[2] N. Nanda, L. Chan, T. Lieberum, J. Smith, and J. Steinhardt, "Progress measures for grokking via mechanistic interpretability," in *International Conference on Learning Representations (ICLR)*, 2023.

[3] A. Gromov, "Grokking modular arithmetic," 2023.

[4] D. Doshi, T. He, A. Das, and A. Gromov, "Grokking modular polynomials," in *ICLR 2024 Workshop on Bridging the Gap Between Practice and Theory in Deep Learning (BGPT)*, 2024.

[5] Z. Liu, O. Kitouni, N. Nolte, E. J. Michaud, M. Tegmark, and M. Williams, "Towards understanding grokking: An effective theory of representation learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[6] X. Davies, L. Langosco, and D. Krueger, "Unifying grokking and double descent," in *NeurIPS 2022 ML Safety Workshop*, 2022.

[7] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.