INITIAL SPECTRAL POWER PREDICTS FREQUENCY SELECTION IN GROKKING AND STEERS IT IN TIED MLPS

Anonymous authors
Paper under double-blind review

ABSTRACT

Networks that grok fit their training data early and generalize much later, using a handful of Fourier frequencies picked up along the way. We ask what decides which frequencies a network ends up with, and whether the same factors decide when it generalizes. We study small networks trained on modular arithmetic, fix our decision rules before running each experiment, and re-test the main results on fresh seeds. For the first question, the initialization matters a great deal. The frequencies with the most initial power in the embedding recover 4 to 13 times more of the final frequencies than chance, in bottlenecked MLPs on two tasks and in a one-layer transformer, and they do better when the initial leaders stand out. In tied MLPs, editing that power steers the outcome in a graded way: an outsider given 0.5, 0.75, 1 or 1.25 times the leader's power wins in 0, 13, 60 and 80% of networks, and a leader whose power is cut never wins. The same edit was not confirmed in the transformer, where it is a weaker lever. Phase alignment, which decides the equal-magnitude case studied in prior theory, carries information only once all powers are made equal. The outcome survives rounding, data order and weight noise, but not a change of data split. Timing behaves differently. Enlarging the winners' head start has no detectable effect, removing all initial contrast roughly doubles the time to grok in an exploratory comparison, and only one predictor survives confirmation: how late the network memorizes. In these full-batch runs, training never climbs the loss walls that straight lines between memorizing and generalizing solutions cross.

1 INTRODUCTION

Grokking is the delayed generalization first reported by Power et al. (2022): a small network fits its training set within a few hundred steps, stays at chance on the test set for thousands more, and only then generalizes. For modular addition we now understand much of what the network learns. It represents residues with a few Fourier frequencies and combines them into a circuit that computes the sum (Nanda et al., 2023; Gromov, 2023; Zhong et al., 2023). Two questions remain open. Which frequencies does a given network end up using, and what decides this? And what sets the long, variable delay before it generalizes?

Existing work offers partial answers. Dynamical accounts disagree on whether generalization is a drift along a connected set of solutions (Boursier et al., 2025; Musat, 2025) or an escape from a metastable memorizing phase (Rubin et al., 2024; Ersoy & Wiesner, 2026). For two-layer networks, He et al. (2026) show that frequencies compete inside each neuron and that the winner depends on both its initial magnitude and its phase alignment; their formal result isolates phase by giving all frequencies the same initial magnitude. More broadly, rich-get-richer theories of feature learning predict that initial conditions break ties between equally useful features (Saxe et al., 2014; Kunin et al., 2025).

We take an experimental approach. Across several campaigns and more than a thousand small networks trained with AdamW and weight decay, we fixed predictions and decision rules before each run (Appendix D), re-tested the main positive findings on seeds never used before, and compared every intervention with an unedited twin or with a control of matched strength. We call a test

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

registered when its decision rule was fixed before its data. Full-batch training is deterministic, so the outcome is trivially a function of the initialization and the data. What we ask is whether a simple, interpretable property of the initialization predicts the outcome, whether editing that property steers it, and whether the same is true of timing. We find the following.

- **Initial power predicts which frequencies win, and in tied MLPs it steers them** (Section 3). On fresh seeds, the highest-power frequencies recover 4–13 times more of the final winners than chance, in bottlenecked MLPs on two tasks and in a one-layer transformer, and prediction improves when the initial leaders stand out. Editing the initial power of tied MLPs moves the outcome in a graded, monotone way. Phase matters only once powers are equalized. The outcome is robust to rounding, data order and weight noise, but depends on the data split.
- **Training does not climb the straight-line walls** (Section 4). Straight lines between memorizing and generalizing solutions cross high loss walls, but training rises 50 to 100 times less, and no more than in dense networks, whose straight lines show no wall at all.
- **Timing is sensitive to the initial spectrum but not accelerated by a head start** (Section 5). Enlarging the winners' head start has no detectable effect on when a network groks, and removing all initial contrast slows it about twofold (exploratory). Neither the closeness of the race, nor barriers, nor parameter divergence explains the spread; the memorization step tracks it, modestly.

Table 1 lists each claim with its evidence and says whether it was registered in advance or found afterwards. Along the way we also developed a symmetry-based progress measure. It turned out to track the confidence of the predictions largely by construction, and we report it in Appendix C.

2 SETUP

Tasks and models. We train on modular addition, subtraction and multiplication with a prime modulus p (97 for MLPs, 113 for transformers); Appendix C also uses composition in the symmetric group S_5. Each task's table is split at random into a training fraction f (0.4 unless stated; 0.3 for transformer addition) and a test set. All models train with full-batch AdamW (learning rate 10^{-3}, $\beta = (0.9, 0.98)$) on cross-entropy. The MLPs map token features $E_a, E_b \in \mathbb{R}^{p\times d}$ to $h = \sigma(E_a[a] + E_b[b])$ and logits hW_2, with either $\sigma(x) = x^2$ ($d = 128$, weight decay 0.3; Gromov, 2023) or ReLU ($d = 256$, weight decay 1). Bottlenecked variants pass both operands through a width-d_e embedding, either *tied* ($E_a = UW_a$ and $E_b = UW_b$ with a single $U \in \mathbb{R}^{p\times d_e}$) or untied. The transformer is the one-layer model of Nanda et al. (2023), without LayerNorm. Dense and tied MLPs also start at different scales (their effective embeddings have variance $1/2d$ and $1/d_e$; Appendix B), so comparisons between them compare whole configurations. We train many seeds in parallel on one GPU. The number of seeds per batch can change the GPU kernels and hence rounding, so we only compare grokking times between runs batched the same way.

Which frequencies win. We read a network's frequencies from its function, not from its weights. For addition, the functional power of frequency k is $P_k = |\hat{L}(k, k, -k)|^2 + |\hat{L}(-k, -k, k)|^2$, and for subtraction $P_k = |\hat{L}(k, -k, -k)|^2 + |\hat{L}(-k, k, k)|^2$, where $\hat{L}$ is the three-dimensional Fourier transform of the class-centered logit table and k runs from 1 to 48 for $p = 97$ and to 56 for $p = 113$. It does not depend on how the network is parameterized. Its concentration is $n_{\text{eff}} = 1/\sum_k \pi_k^2$, with $\pi_k = P_k/\sum_j P_j$. A network's *winners* are its K frequencies with the largest final P_k, where K is the rounded median n_{eff} of its condition within the same set of runs (Appendix B). On the input side, the *initial power* $E_k(0)$ is the summed squared Fourier coefficient at frequency k of both operands' initial effective embeddings (UW_a and UW_b in tied networks, the token embedding in the transformer), and the *predicted winners* are the K frequencies with the largest initial power. Following He et al. (2026), we also define a *ticket* for each unit and frequency, which combines the magnitudes of the unit's input and output weights at that frequency with the cosine of their phase misalignment (Appendix A).

Paths and penalties. The barrier between two solutions is the highest training loss along a path between them, minus the higher of the two endpoint losses. We evaluate straight paths both raw

Table 1: Claims, evidence and status. Registered: decision rule fixed before the data; exploratory: examined after the data.

Claim	Networks	Status
Initial power predicts winners (bottlenecked MLPs, addition)	110 existing, 80 fresh	pre-specified baseline; confirmed on fresh seeds
... in subtraction; in the one-layer transformer	40; 20 fresh	registered; confirmed
Prediction rises with the initial power margin	190	exploratory
Power swap steers winners (tied MLPs)	60 edited, 60 twins	registered; confirmed at threshold
Win rate rises with the edited power ratio	30 parents $\times$ 4 doses	registered; confirmed
Phase predicts winners once powers are equal	40	registered, at threshold; phase-free ticket at chance (exploratory)
Phase changes the winner at fixed power	60 $\times$ 2 arms	registered; inconclusive
Winners survive data order / a new split	20 $\times$ 3 / 20 $\times$ 3	registered; confirmed / intermediate
Power swap steers transformer winners	16	registered; not confirmed
Head start speeds grokking	60	registered; inconclusive
Memorization step tracks grokking time	110, 80	registered; confirmed
Weight norm at memorization tracks grokking time	110, 80	exploratory
Equalized initial power slows grokking	40 vs 40	exploratory
Training does not cross straight-line walls	29 trajectories, 30 pairs	descriptive

and after aligning the MLP's exact reparameterization symmetries (unit permutations, signs and rescalings; Ainsworth et al., 2023), and curved paths, Bézier and polygonal curves optimized for low loss. Spectral penalties add $\alpha \sum_k q_k^2$ to the loss, where q_k is the embedding's share of Fourier power at frequency k; $\alpha < 0$ concentrates the spectrum and $\alpha > 0$ disperses it. We compare each penalty with a *sham*: the same penalty computed after a fixed random permutation of the token axis, which acts with matched strength on a meaningless basis.

Edits, twins and noise. To test causation we edit the initial tied embedding U in its Fourier domain, leaving $\|U\|$ and every other weight unchanged. An edit can swap or rescale the magnitudes of chosen frequencies while keeping their phases, rotate one frequency's row by a common phase while keeping its magnitudes, or make the initial powers of all frequencies equal (Appendix B). Each edited network is compared with its unedited *twin*: the same seed and split, trained without the edit, which reproduces bit for bit. To test robustness we change the number of seeds per batch, add Gaussian noise to every initial weight, train with minibatches of 512 on three data orders, or change the data split. The agreement between two networks' winner sets W and W' is $|W \cap W'|/K$.

3 Initial power predicts and steers the outcome

Prediction at initialization. Our first registered prediction used the ticket: in bottlenecked networks, the K frequencies with the largest summed tickets at initialization should contain the final winners at least three times more often than chance. They do. On 110 existing networks, trained in earlier experiments, the predicted set contains 27.1% of the final winners, against 3.6% by chance (7.5-fold); on 80 networks trained for this test it contains 30.0%, against 4.2% (7.2-fold; exact $p = 7 \times 10^{-28}$ and 4×10^{-29}, Appendix B; per-condition results in Table 4). Initial power alone, which we reported alongside as a pre-specified baseline, did better: 35.5% and 52.5% (9.9- and 12.6-fold). The ticket's phase factor was hurting it. We therefore registered initial power as the predictor and tested it where it had never been measured, again on fresh seeds (Figure 1a). It recovered 45.0% of the winners for subtraction in the same bottlenecked MLP (10.8-fold) and 23.3% in the one-layer transformer on addition (4.4-fold), both at the floor of the permutation test ($p = 10^{-4}$).

Three further checks, made after the fact, point the same way. First, with no choice of K at all, the final winners sit on average at the 92nd percentile of initial power. Second, prediction improves when the initial leaders stand out: across the 190 bottlenecked networks, hit rates rise from 31% to 42% to 55% across terciles of the relative margin between the K-th and $(K+1)$-th initial powers. The spread of initial powers shrinks as the embedding widens (its coefficient of variation is 0.50 at $d_e = 4$, 0.10 at $d_e = 128$ and 0.06 in dense networks), which accounts for the stronger prediction in narrow embeddings, the weaker one in the flatter transformer, and the near-chance prediction in dense networks. Fold-over-chance ratios are not comparable across different K, so we compare conditions through these margins rather than through the ratios. Third, reading the winners when grokking becomes sustained, instead of at the end of training, gives the same hit rates (52.5% and 30.6%); the two winner sets agree at 0.87.

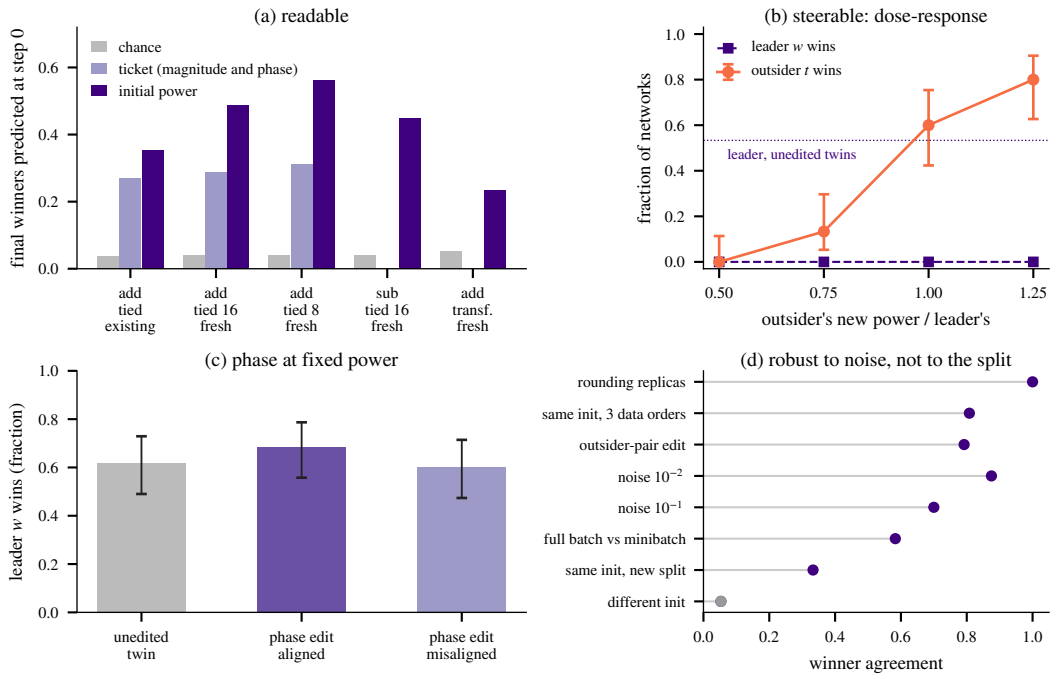

Figure 1: Initial power predicts and steers the outcome. (a) Fraction of final winners predicted from the initial weights by the registered ticket (magnitude and phase) and by initial power; existing and fresh networks, subtraction, and the one-layer transformer. (b) Tied $d_e = 16$ networks whose outsider frequency t receives r times the leader's initial power while the leader w is cut to the outsider's; 30 networks per dose, 95% Wilson intervals. (c) Rotating the leader's Fourier row by a common phase, magnitudes and power unchanged, to raise or lower its ticket; 60 networks each. (d) Agreement of final winners with a reference network under perturbations; gray: different initializations. In (b) the dotted line is the leader's win rate in the unedited $d_e = 16$ twins (53%; 62% pooled over both widths).

Steering by editing. If initial power biases the outcome, changing it should move the outcome, and a larger change should move it more. In tied MLPs this is what happens (Figure 1b). Swapping the magnitudes of the leading frequency w and of a mid-ranked outsider t, with their phases untouched, made t a final winner in 36 of 60 networks and w a winner in none. In the unedited twins, t never won and w won in 62% (exact McNemar $p = 3 \times 10^{-11}$; 60% was exactly the threshold we had set). We then gave the outsider r times the leader's initial power and cut the leader to the outsider's. The outsider won in 0, 13, 60 and 80% of networks at $r = 0.5$, 0.75, 1 and 1.25, and the leader never won at any dose (Cochran–Armitage $p = 5 \times 10^{-13}$; 2×10^{-12} without the $r = 1$ arm, which was built slightly differently). The same 30 parent networks were edited at every dose, with $K = 2$ throughout; permuting doses within parents gives $p = 10^{-4}$, and the response is monotone in 29 of the 30. Because the outsider is chosen among frequencies that do not win in the twin, its base rate is zero by design. Finally, raising the two predicted winners' magnitudes by half made both of them winners in 85% of networks, against 27% of twins.

The role of phase. Why does adding phase make the ticket worse? At initialization the phase misalignments D_{jk} are spread uniformly across units: Rayleigh tests reject uniformity for 5.6% of frequencies, the rate expected by chance. Summed over units, the ticket is therefore initial power plus phase noise. Its phase-averaged version tracks initial power closely (median Spearman correlation 0.96), whereas the ticket itself tracks it at 0.64. Phase should then matter only when power cannot tell frequencies apart, and this is what we find. When every initial power is set exactly equal, the ticket predicts winners at three times chance (12.5% against 4.2%, $p = 0.0015$, just meeting the threshold we had set), while the same ticket without its phase factor recovers only 1.3%, below chance. The remaining information comes from phase, as the equal-magnitude theorem of He et al.

(2026) implies. At fixed power the evidence is weaker. Rotating the leader's Fourier row by a common phase changes none of its magnitudes but raises or lowers its ticket by 12–15%; this moved its win rate from 60% to 68% (Figure 1c; difference $+0.08$, 95% interval $[-0.03, 0.20]$, McNemar $p = 0.27$). We did not detect an effect, and the interval does not rule out one of the size we registered. An earlier edit that scrambled the leader's phases coordinate by coordinate also lowered its effective power (median -9%) and changed winner sets more often than a matched edit of two outsiders' powers (identical sets in 20 against 36 of 60, $p = 0.003$), so we do not read it as a pure phase effect.

Robustness. The outcome is not a fragile by-product of deterministic training (Figure 1d). Replicas that differ only in floating-point rounding end with identical winners in all 24 parents tested (72 comparisons), even though their raw parameters drift apart by 23–40% of their norm; part of that drift may lie along exact symmetries of the factorization. Trained with minibatches on three data orders, an initialization keeps its winners (agreement 0.81, against 0.05 between different initializations). A matched edit of two outsiders' powers keeps 0.79 of them, and Gaussian noise on every weight lowers agreement only gradually, to 0.70 at a tenth of each weight's scale. The data split matters more. The same initialization trained on three new splits agrees with itself at only 0.33 (intermediate, by the rule we had fixed), and full-batch and minibatch runs of one initialization agree at 0.58. Yet initial power still recovers 47% of the winners on the new splits. The initialization favors a set of frequencies, and the split and the training regime choose among them.

Single units, and when the race is decided. In dense networks the per-neuron lottery partly survives: 16–23% of units end on the frequency with their best initial ticket, compared with 10–14% predicted by initial power and 2% by chance. In bottlenecked networks the per-unit ticket falls below initial power (7% against 14%). With dense logging, the winners of a tied network are settled at or before memorization in 17 and 20 of 40 fresh runs ($d_e = 16$ and 8), and at or before grokking in 25 and 28 of 40.

The transformer. We registered the same power swap in the transformer's token embedding, and it was not confirmed: the outsider won in 2 of 16 runs, and in none of the twins (McNemar $p = 0.5$). The transformer's initial spectrum is flatter, so the swap changes the two powers by only about 22%, compared with about 70% in the tied MLPs. Cutting the leader did remove it: the leader lost its place in all six seeds where it had won (exact McNemar $p = 0.03$, not registered). Beyond tied MLPs, steering remains to be established.

4 The path from memorization to generalization

When neurons share a narrow embedding, straight lines between solutions cross high walls (Figure 2a). Between independently trained tied networks the barrier is 10.8 nats at $d_e = 16$ and 15.3 at $d_e = 8$. After aligning unit permutations, signs and rescalings, 4.7 and 10.1 nats remain. These are upper bounds, because we did not align the continuous change of basis of the shared embedding ($U \mapsto UQ$, $W_a \mapsto Q^{-1}W_a$, $W_b \mapsto Q^{-1}W_b$). Untied networks start at 0.82 nats and align down to 0.05. The same holds within a single run: the straight chord from a tied network's memorizing checkpoint to its own final solution crosses 11.2 nats at $d_e = 16$ (range 6.2–24.6) and 9.9 at $d_e = 8$, and test accuracy at its midpoint is only 0.10–0.22. In one-layer transformers, straight lines between independent solutions cross 4.4–4.8 nats, and aligning residual rotations, heads and units does not reduce this; we did not search for curved paths there.

These walls are a property of straight lines, not of the set of solutions. Curved paths optimized in the networks' own parameterization have barriers of 2.5×10^{-5} and 4.8×10^{-4} nats between independent tied solutions, at the resolution of our sampling, and zero between each network's memorizing checkpoint and its own final solution in all three configurations. Their endpoints are preserved to 10^{-3}, and they are only 16–18% longer than the straight line. More importantly, training itself does not climb the walls (Figure 2b). Along the actual trajectory from memorization to the final solution, the training loss rises by at most 0.35 nats (medians 0.19 and 0.12). That is 51 and 103 times less than on the chords (medians of per-network ratios), and no more than in dense untied networks (median 0.22), whose straight lines show no wall at all. Since the trajectory is itself a low-loss path, the flat curve from the memorizing checkpoint adds little; the informative comparison is between

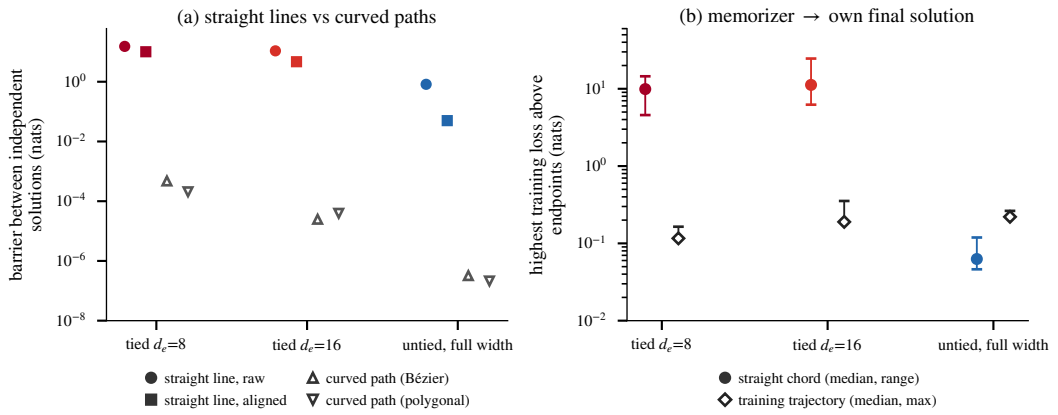

Figure 2: No barrier. (a) Barrier between independently trained solutions (five pairs per condition): straight lines, raw and after aligning the reparameterization symmetries, against optimized curved paths. (b) From each network's memorizing checkpoint to its own final solution: the barrier on the straight chord (median, range) against the highest loss the training trajectory actually reached (median, maximum).

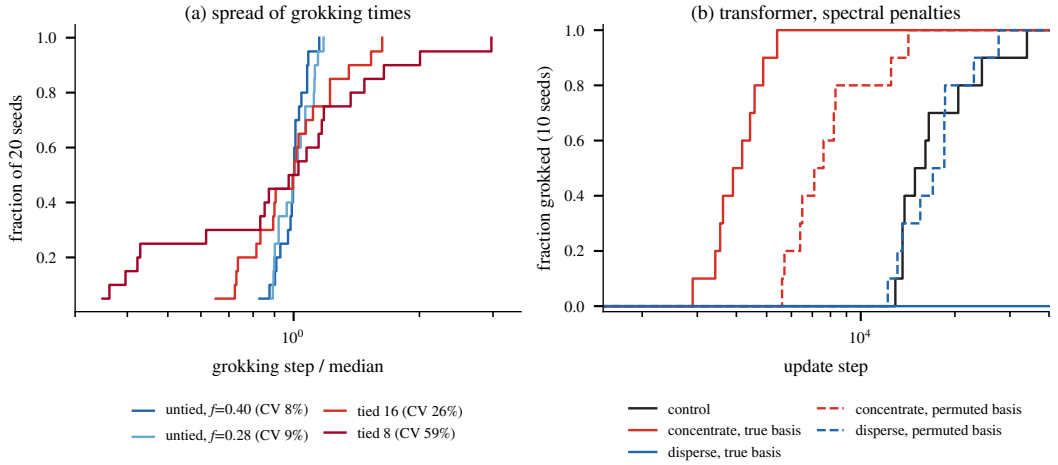

Figure 3: Timing. (a) Grokking steps over 20 seeds, normalized by their median, for untied and tied quadratic MLPs. (b) Transformers under spectral penalties, ten seeds each: fraction grokked by each step, in the true Fourier basis (solid) and in a permuted basis of matched strength (dashed).

trajectory and chord. Measured in training cross-entropy and under full-batch training, grokking in these networks crosses no barrier larger than the ordinary loss excursions of a dense run.

5 The Timing of Grokking

Timing depends on the configuration. Untied quadratic MLPs grok like clockwork. Over 20 seeds, the coefficient of variation of the grokking step is 7.6%, and still 9.4% at a smaller training fraction that makes the delay six times longer (Figure 3a). Tied bottlenecks spread out, with a long right tail: 25.5% at $d_e = 16$ and 58.9% at $d_e = 8$, although all 80 networks grok. Bottlenecked networks also relapse: 56 of 110 existing networks and 32 of 80 fresh ones fall back below 95% test accuracy after first reaching it, which never happens in dense networks.

A noise floor for timing. Rounding alone does not move grokking. Training 19 instead of 20 seeds per GPU batch leaves every network bit-identical, and training one seed per batch, which changes

GPU kernels and rounding, moves grokking steps by a median of 0% (at most 6%), with identical winners in 21 of 21 cases.

Edits move the timing, but not in one direction. Edits of the initial spectrum, by contrast, do move it. Swapping the powers of two outsiders shifts the grokking step by a median of 10%, and by more than 25% in a third of networks. Raising the future winners' power shifts it by a median of 35%, but in both directions: the median signed change is -5.2% (95% interval $[-18\%, +11\%]$, Wilcoxon $p = 0.65$; bootstrap interval $[-20\%, +6\%]$ when grokking is dated by a sustained crossing), and the classification we had fixed in advance is inconclusive. Three statistics of initial power also failed a registered selection test as predictors of timing (ρ from -0.14 to -0.24 on existing networks), although they kept the same sign on fresh ones (down to -0.30).

Removing the contrast slows grokking. The only large effect of the initial spectrum on timing is a global one. Networks whose initial powers were all set equal grokked at a median of 16,000 steps, against 8,500 for unedited networks of the same configuration with other seeds (Mann–Whitney $p = 2 \times 10^{-10}$; exploratory). Removing all initial contrast slows grokking; enlarging an existing head start does not detectably speed it up.

Not the closeness of the race. If the outcome is a race that starts at initialization, a close race might take longer. We registered this as a stratified Spearman correlation of at most -0.3 between the grokking step and the initial gap separating the last winner from the first loser. The prediction failed in both sets, with $\rho = +0.073$ on existing networks and $+0.078$ on fresh seeds ($p = 0.76$, 95% interval $[-0.14, 0.28]$), under every reading of the statistic we examined.

Not barriers or parameter divergence alone. No barrier is crossed (Section 4), and divergence in parameter space does not produce the spread either. A dense ReLU MLP without any bottleneck diverges more under rounding than any bottlenecked network (to 64% of its norm; 24% at the bottlenecks' weight decay), yet over ten seeds its grokking steps stay between 7,100 and 8,400. The dense quadratic MLP, by contrast, stays within a factor of about 17 of the round-off floor. Parameter divergence can coexist with clock-like timing, and in every configuration the replicas end with identical winners and test accuracies within 10^{-3}.

What tracks the timing. Of four candidate predictors we registered, one passed selection on the existing networks and confirmation on fresh ones: the memorization step. Networks that memorize later grok later ($\rho = +0.35$ in both sets, Holm $p = 0.0014$ on fresh seeds; $+0.39$ when grokking is dated by a sustained crossing). This is not simply because grokking time includes memorization time: networks that memorize later also wait longer after memorizing ($\rho = +0.34$). In an exploratory analysis, the weight norm at memorization does as well in both sets ($\rho = +0.40$ and $+0.37$ with the grokking step, $+0.39$ and $+0.35$ with the delay after memorization), in line with accounts in which weight decay sets the clock of grokking (Lyu et al., 2024; Boursier et al., 2025). Each accounts for about a tenth of the rank variation.

What moves the timing. Four interventions change the timing, and the sham controls separate what depends on Fourier structure from what does not.

- *A concentration pulse* of 1,000 updates during the plateau of the untied quadratic MLP advances grokking by 200 steps relative to its sham (95% interval $[-245, -168]$; 20 parents, $p = 10^{-4}$), about 7% of the grokking time. The effect replicated on ten fresh parents (-237.5 steps, against -187.5 on the first ten).
- *Sustained concentration* makes the transformer grok 3.8 times faster (median 3,900 against 14,900 steps; every penalized run groks before every control, Figure 3b). About 70% of this speed-up (95% interval 56–82%) survives in the permuted basis. The remaining, basis-specific part is 3,263 steps $[2{,}150, 4{,}750]$, and every true-basis run again groks before every sham run.
- *Sustained dispersion* is where the basis does all the work. In the true basis it stops the transformer from grokking in all ten runs within 40,000 steps. In the permuted basis, with the same penalty value and 20% more cumulative gradient, it is indistinguishable from no penalty (median 17,000 against 14,900 steps; log-rank $p = 0.93$).

- *Concentration in MLPs* stops the ReLU MLP (10/10) and the untied quadratic MLP (6/6) from ever generalizing, but the permuted-basis penalty does exactly the same at matched strength, over ten and six independent permutations. It collapses the rank of the embedding and says nothing about Fourier structure.

Concentration therefore changes grokking mostly through generic effects on the embedding, whereas dispersion in the true basis is the only spectral lever whose effect depends on the Fourier basis itself.

6 Related work

Mechanisms of grokking. Networks trained on modular arithmetic build Fourier circuits (Nanda et al., 2023; Gromov, 2023; Chughtai et al., 2023; Zhong et al., 2023; Morwani et al., 2024), and generalization has been linked to representation learning (Liu et al., 2022), weight norm (Liu et al., 2023), circuit efficiency (Varma et al., 2023), a transition from lazy to rich training (Kumar et al., 2024; Lyu et al., 2024), label noise (Doshi et al., 2024) and numerical effects (Prieto et al., 2025). Closest to our work, He et al. (2026) show that each neuron's frequency is a lottery won at initialization through magnitude and phase alignment, with experiments under random initialization and AdamW, and prove that phase decides when all frequencies start with equal magnitudes. We work at the population level: initial power predicts the outcome on fresh seeds and across architectures, editing it steers the outcome with a dose-response, phase regains a weak predictive role once powers are equalized, and the per-neuron lottery survives only partly, in dense networks. Rich-get-richer dynamics in factorized parameterizations, where modes are learned in order of strength and ties at small initialization are broken by initial conditions, are well studied (Saxe et al., 2014; Li et al., 2021; Jacot et al., 2021). For quadratic networks on modular addition, Alternating Gradient Flows predicts that Fourier features are acquired in decreasing order of their coefficients; with one-hot inputs these are all equal, so the initialization must break the tie (Kunin et al., 2025). Our experiments measure how initial power breaks it, and test that causally. Lottery-style accounts of which structures training keeps go back to Frankle & Carbin (2019); in grokking, sparse and dense subnetworks compete (Merrill et al., 2023), and lottery-ticket subnetworks can accelerate generalization (Minegishi et al., 2023).

Progress measures. Hwang & Park (2026) track violations of commutativity and associativity and accelerate grokking by penalizing them, while Sivasankar (2026) and Khanh et al. (2026) track Fourier synchronization and the spectral entropy of internal representations. Our own output-based measure (Appendix C) is, for simply transitive actions, close to a test cross-entropy with symmetry-transported labels. Its peak reflects confidence, and we recommend confidence controls for measures of this kind.

Dynamics and loss landscapes. Our trajectories are consistent with drift along connected solutions (Li et al., 2022; Boursier et al., 2025; Musat, 2025) in these full-batch networks. They do not test noise-driven escape over barriers of the regularized objective (Rubin et al., 2024; Ersoy & Wiesner, 2026), which requires stochastic training. They complement a singular-learning-theory account of basin selection (Cullen et al., 2026), and the exploratory link between the weight norm at memorization and the delay fits accounts in which weight decay sets the clock (Lyu et al., 2024; Boursier et al., 2025). That curved paths connect solutions separated by straight-line walls is familiar from the mode-connectivity literature (Garipov et al., 2018; Draxler et al., 2018; Frankle et al., 2020; Entezari et al., 2022; Ainsworth et al., 2023). What is new here is the pair we connect, a memorizing and a generalizing solution, and the training trajectory itself, which never climbs. Khanh (2026) recombines weights across grokking runs, and Tang et al. (2026) report topological signatures whose visibility, in our networks, follows the concentration of the embedding.

7 Limitations and conclusion

Our models are small and our tasks are modular arithmetic. Steering is established only in tied quadratic MLPs. In the transformer, where the swap moved powers by about 22% instead of 70%, it was not confirmed, and we did not test a stronger dose. Embedding width, the spread of initial

powers and the initialization scale vary together across our conditions, so the width dependence of prediction is not separated from scale. The outcome depends on the data split and on the training regime. Initial power biases the outcome strongly but recovers only about half of it, and K is set for each set of runs from their final concentration; the percentile and margin analyses avoid that choice. Several registered tests passed exactly at thresholds set on point estimates (the power swap, the data-order agreement and the equal-power ticket), and the tests of phase at fixed power and of the head start's effect on timing were inconclusive. The barrier results use training cross-entropy, full-batch training and MLPs only, and the aligned straight-line barriers are upper bounds. All training uses AdamW with weight decay, whose per-coordinate normalization may change how magnitude and phase compete; a gradient-descent control is the most important experiment we have not run. Relapses after grokking may be related to adaptive-optimizer spikes (Thilak et al., 2022). Finally, the timing of grokking remains largely unexplained.

In the networks we studied, the frequencies a network ends up using are strongly biased toward those its initial weights gave the most power, and in tied MLPs, moving that power moves the outcome in proportion. The time at which it generalizes is sensitive to the same spectrum, is not shortened by boosting the winners, and tracks, modestly, how late and how heavily the network memorizes.

AI USE STATEMENT

In this work, we used generative AI tools for literature search, research ideation, experiment design, writing and running code (including with autonomous coding agents), data analysis, and drafting of the manuscript. The authors set the research questions and evaluation criteria, approved each pre-registration before it was frozen, decided scope and claims, and reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments train small models from scratch on synthetic tasks, with fixed seeds and recorded batch sizes. The anonymized supplementary material contains the code, every registration file with its hash, per-run configurations, logs and results, the final weights of the fresh-seed confirmatory runs, a script that regenerates their initial weights from the seeds and checks them, and each campaign's report and audit. Appendix A defines every measure, Appendix B gives initialization, penalty, path and statistical settings, and Appendix D lists the principal registered predictions with their outcomes.

REFERENCES

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023.

Etienne Boursier, Scott Pesme, and Radu-Alexandru Dragomir. A theoretical framework for grokking: Interpolation followed by Riemannian norm minimisation. *arXiv preprint arXiv:2505.20172*, 2025.

Bilal Chughtai, Lawrence Chan, and Neel Nanda. A toy model of universality: Reverse engineering how networks learn group operations. In *International Conference on Machine Learning*, 2023.

Ben Cullen, Sergio Estan-Ruiz, Riya Danait, and Jiayi Li. A basin-selection perspective on grokking via singular learning theory. *arXiv preprint arXiv:2603.01192*, 2026.

Darshil Doshi, Aritra Das, Tianyu He, and Andrey Gromov. To grok or not to grok: Disentangling generalization and memorization on corrupted algorithmic datasets. In *International Conference on Learning Representations*, 2024.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International Conference on Machine Learning*, 2018.

Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. In *International Conference on Learning Representations*, 2022.

Ibrahim Talha Ersoy and Karoline Wiesner. Noise-driven escape from metastable phases explains grokking in deep neural networks. *arXiv preprint arXiv:2606.17120*, 2026.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, 2020.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of DNNs. In *Advances in Neural Information Processing Systems*, 2018.

Andrey Gromov. Grokking modular arithmetic. *arXiv preprint arXiv:2301.02679*, 2023.

Jianliang He, Leda Wang, Siyu Chen, and Zhuoran Yang. On the mechanism and dynamics of modular addition: Fourier features, lottery ticket, and grokking. *arXiv preprint arXiv:2602.16849*, 2026.

Hyeonbin Hwang and Yeachan Park. Intrinsic task symmetry drives generalization in algorithmic tasks. *arXiv preprint arXiv:2603.01968*, 2026.

Arthur Jacot, François Ged, Berfin Şimşek, Clément Hongler, and Franck Gabriel. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint arXiv:2106.15933*, 2021.

Truong Xuan Khanh. Cross-trajectory chimera interventions reveal dissociable roles of weight magnitude and direction in grokking. *arXiv preprint arXiv:2607.06628*, 2026.

Truong Xuan Khanh, Truong Quynh Hoa, Luu Duc Trung, and Phan Thanh Duc. Spectral entropy collapse as a phase transition in delayed generalisation: An interventional and predictive framework for grokking. *arXiv preprint arXiv:2604.13123*, 2026.

Tanishq Kumar, Blake Bordelon, Samuel J. Gershman, and Cengiz Pehlevan. Grokking as the transition from lazy to rich training dynamics. In *International Conference on Learning Representations*, 2024.

Daniel Kunin, Giovanni Luca Marchetti, Feng Chen, Dhruva Karkada, James B. Simon, Michael R. DeWeese, Surya Ganguli, and Nina Miolane. Alternating gradient flows: A theory of feature learning in two-layer neural networks. In *Advances in Neural Information Processing Systems*, 2025.

Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021.

Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss? – a mathematical framework. In *International Conference on Learning Representations*, 2022.

Ziming Liu, Ouail Kitouni, Niklas Nolte, Eric J. Michaud, Max Tegmark, and Mike Williams. Towards understanding grokking: An effective theory of representation learning. In *Advances in Neural Information Processing Systems*, 2022.

Ziming Liu, Eric J. Michaud, and Max Tegmark. Omnigrok: Grokking beyond algorithmic data. In *International Conference on Learning Representations*, 2023.

Kaifeng Lyu, Jikai Jin, Zhiyuan Li, Simon S. Du, Jason D. Lee, and Wei Hu. Dichotomy of early and late phase implicit biases can provably induce grokking. In *International Conference on Learning Representations*, 2024.

William Merrill, Nikolaos Tsilivis, and Aman Shukla. A tale of two circuits: Grokking as competition of sparse and dense subnetworks. *arXiv preprint arXiv:2303.11873*, 2023.

Gouki Minegishi, Yusuke Iwasawa, and Yutaka Matsuo. Bridging lottery ticket and grokking: Understanding grokking from inner structure of networks. *arXiv preprint arXiv:2310.19470*, 2023.

Depen Morwani, Benjamin L. Edelman, Costin-Andrei Oncescu, Rosie Zhao, and Sham Kakade. Feature emergence via margin maximization: Case studies in algebraic tasks. In *International Conference on Learning Representations*, 2024.

Tiberiu Musat. The geometry of grokking: Norm minimization on the zero-loss manifold. *arXiv preprint arXiv:2511.01938*, 2025.

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

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Lucas Prieto, Melih Barsbey, Pedro A. M. Mediano, and Tolga Birdal. Grokking at the edge of numerical stability. In *International Conference on Learning Representations*, 2025.

Noa Rubin, Inbar Seroussi, and Zohar Ringel. Grokking as a first order phase transition in two layer networks. In *International Conference on Learning Representations*, 2024.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations*, 2014.

Achyuthan Sivasankar. Circuit synchronization precedes generalization: A causal precursor to grokking. *arXiv preprint arXiv:2606.12966*, 2026.

Yifan Tang, Qiquan Wang, Inés García-Redondo, and Anthea Monod. Topological signatures of grokking. *arXiv preprint arXiv:2605.06352*, 2026.

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

Vikrant Varma, Rohin Shah, Zachary Kenton, János Kramár, and Ramana Kumar. Explaining grokking through circuit efficiency. *arXiv preprint arXiv:2309.02390*, 2023.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. The clock and the pizza: Two stories in mechanistic explanation of neural networks. In *Advances in Neural Information Processing Systems*, 2023.

A MEASURES

Tickets. For unit j and frequency k, let $\hat{A}_{jk} = \sum_a E_a[a, j]e^{-2\pi ika/p}$, $\hat{B}_{jk}$ likewise for E_b, and $\hat{C}_{jk} = \sum_c W_2[j, c]e^{-2\pi ikc/p}$. The misalignment is $D_{jk} = \arg \hat{A}_{jk} + \arg \hat{B}_{jk} - \arg \hat{C}_{jk}$ wrapped to $(-\pi, \pi]$, and the ticket is $T_{jk} = (|\hat{A}_{jk}||\hat{B}_{jk}||\hat{C}_{jk}|)^{1/3} \cos D_{jk}$. The population ticket is $T_k = \sum_j \max(T_{jk}, 0)$ at initialization; the predicted winners are the K largest T_k. The number of winners K is the median final functional concentration of the condition, rounded. Initial weights are regenerated from the seed and verified bit-exact against stored step-0 weights. The race-closeness statistic is $g = (T_{(K)} - T_{(K+1)})/T_{(K)}$ over the ordered population tickets; its correlation with the grokking step is computed on ranks within each condition, pooled, with a within-condition permutation test.

Table 2: The arc in every run. Peak: maximum naturality violation (nats). Steps are the checkpoint of the peak, the first step with training accuracy $\geq 99\%$ (memorized), and the first steps with test accuracy $\geq 50\%$ and $\geq 95\%$. Collapse: peak divided by the value at the end of training. Ranges are over seeds (three per cell; four for S_5). $^*S_5$ was still falling at the 100k-step budget.

Architecture	Task	Arcs	Peak	Peak step	Memorized	50% test	95% test	Collapse
Quadratic MLP	$a+b$	3/3	5.85–6.09	993	440	1.3k–1.4k	2.8k–3.1k	752–3255
Quadratic MLP	$a-b$	3/3	5.93–6.98	993–1.2k	440–460	1.3k–1.6k	3.0k–3.5k	1215–1737
Quadratic MLP	ab	3/3	5.23–5.81	915	419–440	1.2k–1.3k	2.6k–2.8k	7055–1.3×10^4
ReLU MLP	$a+b$	3/3	9.05–9.68	2.2k	456–460	4.2k–4.8k	7.4k–8.2k	170–246
ReLU MLP	$a-b$	3/3	8.83–9.39	2.2k	456–480	4.0k–4.4k	6.7k–7.7k	180–225
ReLU MLP	ab	3/3	8.67–9.03	2.2k	456	3.8k–4.2k	6.6k–7.0k	326–507
Transformer	$a+b$	3/3	18.14–18.39	1.6k–2.7k	250–350	11.2k–35.0k	11.8k–36.0k	449–3.2×10^5
Transformer	$a-b$	3/3	20.19–20.44	1.6k–2.4k	400–426	17.9k–24.9k	20.2k–28.1k	202–7.6×10^4
Transformer	ab	3/3	12.34–14.49	1.6k–1.9k	426–529	2.2k–5.6k	3.0k–6.3k	2.4×10^4–2.0×10^6
ReLU MLP	S_5 (3 actions)	12/12	10.97–11.26	1.7k	300	20.1k–23.5k	64.7k–94.5k	11–14*

Training events. Memorized: first evaluation with training accuracy $\geq 99\%$. Grokked: first evaluation with test accuracy $\geq 95\%$. Grokking is *sustained* from the first evaluation after which test accuracy stays at or above 95% for the rest of training. The decision step is the first evaluation from which the top-K set of P_k equals the final winners for the rest of training.

Chaos. Replicas share seed, initialization and split and differ only in the number of seeds batched together, which changes floating-point association order. Divergence is the relative parameter distance between replicas, reported against the round-off floor measured early in training under two conventions (Appendix D files report both).

B Experimental details

Initialization. Untied MLPs draw the entries of E_a and E_b from $\mathcal{N}(0, 1/2d)$ and of W_2 from $\mathcal{N}(0, 1/d)$. Tied MLPs draw U, W_a and W_b from $\mathcal{N}(0, 1/d_e)$ and W_2 from $\mathcal{N}(0, 1/d)$. Each seed draws its tensors in a fixed order on the CPU, so initial weights are reproducible bit for bit. The transformer uses the initialization of the reference implementation in the supplementary code.

Training and evaluation. Bottlenecked MLPs train for 30,000 updates, dense MLPs for 12,000 and transformers for 40,000. In the fresh confirmatory runs, accuracies and P_k are evaluated every 25 updates up to 3,000, every 100 up to 10,000 and every 500 afterwards; elsewhere on geometric schedules of at least 45 checkpoints. Minibatch runs use batches of 512 and the same number of updates.

Spectral penalties. The penalty is $\alpha \sum_k q_k^2$, with q_k the embedding's share of Fourier power at frequency k. Sustained penalties apply throughout training, with $\alpha = -0.1$ for concentration and $\alpha = +0.01$ (transformer) or $+0.001$ (ReLU MLP) for dispersion. The pulse applies $\alpha = -0.03$ for 1,000 updates from a fixed checkpoint of the memorization plateau; its sham branches from the same checkpoint. The sham computes the same penalty after a fixed random permutation of the token axis. Penalty values of true and sham arms were matched within 0.3%, and their cumulative penalty gradients are reported in the supplementary material.

Paths. Curved paths are Bézier curves and polygonal chains (six path families, three restarts each) whose control points are optimized on training loss in the networks' native parameterization. Barriers are evaluated on dense grids along each path, and the endpoints are preserved to within 10^{-3}.

Statistics. Hit rates are tested against random K-subsets of frequencies with 10,000 permutations. Stratified correlations use ranks within conditions and permutations within conditions. Medians of paired relative changes use distribution-free or bootstrap intervals over networks, and proportions use Fisher exact tests and paired bootstrap intervals.

Further edits and statistics. The dose edit rescales the Fourier rows of the outsider and the leader so that the outsider's initial power becomes r times the leader's and the leader's becomes the outsider's, before a global rescale restores $\|U\|_F$; the $r = 1$ arm is the per-column magnitude swap of the earlier campaign. The phase edit multiplies the leader's Fourier row by one common phase, chosen on a grid of 64 values to maximize or minimize its population ticket; it leaves all magnitudes and $E_w(0)$ unchanged to 3×10^{-8}. The equal-power edit rescales every Fourier row so that all $E_k(0)$ are equal. The outsider edit exchanges the magnitudes of the 20th- and 30th-ranked non-winning frequencies. Paired binary outcomes use exact McNemar tests. The exact p-values of the ticket hit rates are computed from the null distribution of hit counts under random K-subsets; the permutation tests reach their floor of 10^{-4}. K is the rounded median n_{eff} of each set of runs, so tied $d_e = 8$ has $K = 1$ among the existing networks and $K = 2$ among the fresh ones. Batch-filler seeds run alongside registered ones are excluded from every registered analysis.

C A SYMMETRY MEASURE THAT MOSTLY READS CONFIDENCE

Symmetry-based measures have been proposed as circuit-free progress measures for grokking (Hwang & Park, 2026). We developed one independently and report it here because it mostly reads confidence.

Naturality. A group G acts on inputs $x = (a, b)$ and answers c so that the true rule is equivariant; for addition, $g = (s, t)$ maps $(a, b, c) \mapsto (a+s, b+t, c+s+t)$. With predicted distribution $P(c \mid x)$ and moved prediction $P^g(c \mid x) = P(g \cdot c \mid g \cdot x)$,

$$\mathcal{N} = \mathbb{E}_g \, \mathbb{E}_x \, D_{\mathrm{KL}}\big(P^g(\cdot \mid x) \,\|\, P(\cdot \mid x)\big), \tag{1}$$

averaged over the full input table and 12 random group elements (Appendix A gives the other groups). $\mathcal{N} = 0$ exactly when the predictions commute with the action; it needs only a forward pass.

One arc everywhere. In every run we measured, $\mathcal{N}$ follows the same course (Figure 4; Table 2). It starts near zero, as it must for near-uniform outputs; it rises by a factor of 230 to 39,000 while the network memorizes; it peaks after memorization and before test accuracy reaches 50%; and it collapses by two to six orders of magnitude as the network generalizes (on S_5 the collapse was still under way when training stopped). The arc appears in all 39 arcs of the architecture–task grid and of S_5, including left, right and inner actions of the non-abelian group, and in all 80 networks of a later confirmatory set.

The peak reads confidence. Because a KL divergence grows with the confidence of the predictions, the peak could measure confidence rather than symmetry. It largely does. Across 80 networks in eight embedding conditions, peak height correlates with the margin between the top two logits at $\rho = +0.954$ and with the prediction entropy at $\rho = -0.940$, but only at $+0.48$ with weight norm. Matching entropy by temperature reduces the spread of peak heights across conditions from 19.8-fold to about threefold. An intervention confirms the coupling: a concentration penalty that stops a quadratic MLP from generalizing raises its peak 4.7-fold while its logits grow 20.6-fold.

The restoration survives. The collapse is more robust. Its raw median over the 27 grid networks is 1,206-fold. At matched entropy it is 9.3-fold, above fivefold in all 27 networks; under logit-RMS normalization it is 3.5-fold, above fivefold in only 5 of 27. Generalizing networks do become more equivariant than their confidence alone explains, by about an order of magnitude.

Commutativity comes first. The transformer on addition reaches 0.27–0.28 test accuracy during its plateau, close to the 0.30 obtained by answering each test pair with its transposed training pair; on subtraction, where transposition does not help, it stays at chance (0.003). The network is largely commutative while its violation of the translation action peaks. Identities of the operation such as commutativity (Hwang & Park, 2026) can be satisfied by a lookup table; equivariance under the group action cannot, and the two separate in time.

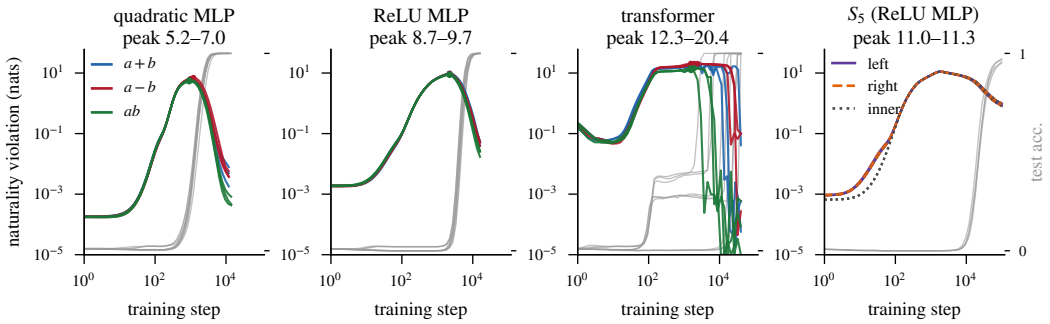

Figure 4: Naturality $\mathcal{N}$ (Equation (1); colored, left axis) and test accuracy (gray, right axis) for every run of the architecture–task grid (three seeds each) and for S_5 (four seeds, three group actions). Dots mark peaks.

A registered link that failed. We predicted that networks with more concentrated spectra would break the symmetry harder (Spearman $\rho \leq -0.7$ between median peak and concentration n_{eff} over eight conditions). With ten seeds per condition the correlation is $+0.43$, and it stays refuted under all four confidence controls. The peak does not read the network's regime, and adding $\mathcal{N}$ to internal diagnostics improves the prediction of grokking by only 0.003–0.010 in average precision.

What the measure sees. A KL-based symmetry measure sees the passage from memorization to generalization, but not which frequencies win or when; its peak should be reported with a confidence control.

Naturality. Logits are centered over answers before the softmax. Equation (1) uses the full table of p^2 inputs (14,400 for S_5) and 12 random group elements; 4, 12 and 48 elements agree within 5%. For subtraction the action is $(a+s, b+t, c+s-t)$. For multiplication, tokens are re-indexed by the discrete logarithm, the action (ua, vb, uvc) becomes a translation, and the token 0 is excluded. For S_5 we use left $(g\sigma, \tau, g\rho)$, right $(\sigma, \tau g, \rho g)$ and inner $(\sigma g^{-1}, g\tau, \rho)$ actions. Confidence controls: entropy matching rescales logits by a temperature that fixes the mean prediction entropy; RMS normalization divides logits by their root mean square.

Why it reads confidence. For addition, the translation action is simply transitive on inputs: a predictor with $\mathcal{N} = 0$ has the form $f(c-a-b)$, and each term of Equation (1) compares the prediction at one input with the transported prediction at another. When one of the two is a training input, the term is approximately a cross-entropy against a label transported from the training set. $\mathcal{N}$ is therefore close to a test cross-entropy with symmetry-transported labels, which explains why its peak tracks confidence and why it collapses at generalization; beyond test loss it adds little.

D REGISTERED PREDICTIONS

Table 3: The principal predictions registered before their data, with their outcomes; the complete registrations, with file hashes and campaign logs, are in the supplementary material (hashes establish integrity, not temporal precedence). For the prediction from initial weights, stage 1 reused existing networks whose grokking times were known but whose initialization statistics had not been computed; stage 2 used fresh seeds.

Prediction	Test	Outcome
Transformers show the naturality arc	arc in $\geq 2/3$ seeds	Confirmed, 3/3 (and 39/39 arcs overall)
Transformer grokking survives 10–20% label noise	grokking by 40k steps	Refuted: no run with $\xi \geq 0.1$ generalizes
Peak height rises with concentration	$\rho \leq -0.7$ over 8 conditions	Refuted: $\rho = +0.43$
Concentration and dispersion transfer across architectures	five pre-registered clauses	All five passed; then qualified by the sham controls below
Parameter divergence rate falls with n_{eff}	$\rho \leq -0.7$ over 8 conditions	Refuted: $\rho = +0.69$
Basis dependence of the MLP concentration failures	classification rule fixed in advance	Generic in both architectures
Basis dependence of the transformer dispersion failure	classification rule fixed in advance	Basis-specific: sham indistinguishable from control
Dense networks stay near the round-off floor	$\geq$ 3 decades below bottlenecks	Refuted by a dense ReLU MLP
Winners readable at step 0	$h \geq 3\times$ chance, $p < 0.01$	Confirmed on existing and fresh networks
Race closeness sets the grokking time	stratified $\rho \leq -0.3$	Refuted twice ($+0.073$, $+0.078$)
No such relation in dense networks	$\lvert\rho\rvert < 0.3$	Confirmed twice (wide intervals)
Initial-power head start predicts the grokking time (three statistics)	selection on existing networks	Not selected (ρ from -0.14 to -0.24)
Memorization step predicts the grokking time	selection, then confirmation on fresh seeds	Confirmed ($\rho = +0.35$ in both sets)
Swapping initial power moves the winner	$\geq 60\%$, $p < 0.01$	Confirmed at threshold (60% vs 0%; McNemar $p = 3 \times 10^{-11}$)
Scrambling phases leaves the winner's odds	change ≤ 0.15	Point estimate within margin ($+0.05$; interval to 0.20); equivalence not established; the edit also changed effective power
Head start steers the grokking time	neutral three-way rule	Inconclusive (median -5%, interval $[-18\%, +11\%]$; same with sustained grokking)
Winners survive minibatch data order	agreement ≥ 0.8 and $\geq 3\times$ between seeds	Confirmed (0.81 vs 0.05)
Initial power predicts winners in subtraction and in the transformer	$h \geq 3\times$ chance, $p < 0.01$	Confirmed ($10.8\times$; $4.4\times$)
Rounding and outsider edits set the noise floor	Mann–Whitney / Fisher rules	Edits exceed the floor (rounding floor exactly zero)
Phase at fixed power changes the winner	paired difference ≥ 0.20	Inconclusive ($+0.08$, interval $[-0.03, 0.20]$)
Ticket predicts winners at equal power	$h \geq 3\times$ chance, $p < 0.01$	Confirmed exactly at threshold ($3.0\times$)
Win rate rises with the edited power ratio	Cochran–Armitage $p < 0.01$	Confirmed (0, 13, 60, 80%)
Winners survive a new data split	agreement ≥ 0.6	Intermediate (0.33)
Power swap steers transformer winners	$\geq 40\%$, McNemar $p < 0.05$	Not confirmed (2 of 16)

Table 4: Prediction from initial weights, per condition. K: number of final winners per network in the condition. h: fraction of final winners predicted at step 0 by the registered ticket, the ticket without its phase factor, and initial power alone. Unit match: fraction of units ending on their own predicted frequency (ticket / initial power); chance $1/48 \approx 0.021$. Stage 1: existing networks (discovery); stage 2: fresh seeds 903–942 (confirmation). †From an earlier study of the spread of grokking times (20 seeds each, independent splits).

Stage	Condition	n	K	Chance	h ticket	h no phase	h power	Unit (ticket)	Unit (power)	Median grok
1	tied 4	10	1	0.021	0.30	0.40	0.40	0.094	0.247	1975
1	tied 8	10	1	0.021	0.30	0.60	0.50	0.077	0.182	8800
1	tied 16	10	2	0.042	0.25	0.30	0.25	0.059	0.109	7450
1	tied 32	10	2	0.042	0.35	0.40	0.35	0.074	0.102	9500
1	untied 16	10	3	0.062	0.33	0.33	0.40	0.052	0.115	18800
1	tied 64	10	2	0.042	0.20	0.35	0.15	0.072	0.076	9200
1	tied 128	10	2	0.042	0.10	0.30	0.25	0.060	0.052	10600
1	untied full	10	24	0.500	0.53	0.53	0.54	0.232	0.126	2938
1	tied 16†	20	2	0.042	0.28	0.42	0.45	0.072	0.128	8225
1	tied 8†	20	1	0.021	0.30	0.35	0.35	0.080	0.189	7450
1	untied, $f = 0.28$†	20	27	0.562	0.59	0.60	0.61	0.162	0.104	15475
1	untied, $f = 0.40$†	20	29	0.604	0.64	0.65	0.66	0.226	0.136	2970
2	tied 16	40	2	0.042	0.29	0.40	0.49	0.067	0.122	8500
2	tied 8	40	2	0.042	0.31	0.46	0.56	0.070	0.171	8550
2	untied full	40	28	0.583	0.63	0.63	0.62	0.228	0.140	2900