WHAT A PLATEAU CAN BE RESCUED BY IS NOT WHAT IT IS SPENT BUILDING: COMPONENT SURGERY IN A GLASS-BOX NETWORK

Anonymous authors
Paper under double-blind review

ABSTRACT

A grokking plateau poses two questions usually conflated: which component, supplied, ends it, and which one training spends it building. We separate them — exploitable from deferred — in a glass-box network whose hidden units all compute one fixed primitive, $e^x - \ln(y)$, and learn only routing and gains, so its component boundaries are architectural. It groks modular addition in 17/19 runs. Implanting a run's 100k-step embedding into its own 30k-step checkpoint reaches 0.9 test accuracy in 1.2k steps against 12.7k for a fresh-optimizer fine-tune-only baseline; a capacity-matched readout head is $\geq 3\times$ slower, a random direction at matched norm does not rescue on this task, and rescaled host structure ends below baseline at 20k. An untrained cosine bank rescues if its frequencies are position-shared. A registered control refuses the missing-component reading: the host already spans 87% of the donor embedding's norm, and that in-span part reads higher at 2k (0.823–0.834 against the orthogonal part's 0.289–0.527, three hosts, though it crosses 0.9 first on only two of them) — yet 72% of it is not the host's own structure rescaled. The orthogonal part rescues too, so the in-span part is better early, not necessary: amplification fits as well as supply. Either way the rescue works only in a host short of the solution. Exploitable and deferred come apart on both tasks — embedding exploitable, wiring deferred — but the *wiring's* exploitability inverts: harmful early on modular addition, immediately useful on parity. What parity's embedding defers is its allocation of power to the support bits, not its convergence. All results are small-scale, on two task families.

1 INTRODUCTION

Grokking (Power et al., 2022) is well documented but very little dissected. We know about when a network groks (Liu et al., 2023; Nanda et al., 2023; Gromov, 2023; Barak et al., 2022) but not enough about what a plateau is spent on. Two questions hide inside that one: which component, supplied, ends the plateau — the *exploitable* one — and which component training spends the plateau building — the *deferred* one. Existing papers watch the training and correlate, or reshape a run before it begins — hyperparameters (Liu et al., 2022; 2023), masks (Minegishi et al., 2025), embeddings at initialization (Xu et al., 2025), norm/direction chimeras across independently seeded runs (Truong, 2026a). None, so far as we know, have implanted a run's own future structure into its own past self to ask which component ends the plateau. The nearest procedural relative is weight rewinding (Frankle et al., 2020), which also uses one run's own history but moves the weights *backwards*, carries a mask rather than a component, and asks about trainable sparsity (G).

The substrate is a network that is readable. Hidden units compute $\mathrm{eml}(x, y) = e^x - \ln y$ and learn only which arguments to select and with what gain. The network therefore factors as Figure 1a shows, and the *wiring* prints as a symbolic program (Section 2). The print is of the routing: the embedding does not appear in it, so the 98.95% agreement measures chooser discretization. The substrate is not the story: printing is a translator, not a builder.

The headline intervenes on a single run's own history: the embedding of a grokked run at 100k steps is transplanted into the same run's past self at 30k and fine-tuned for 2k steps. The host reaches 0.932 against the 0.240 fine-tune-only baseline — a roughly tenfold compression on the one reset-

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.

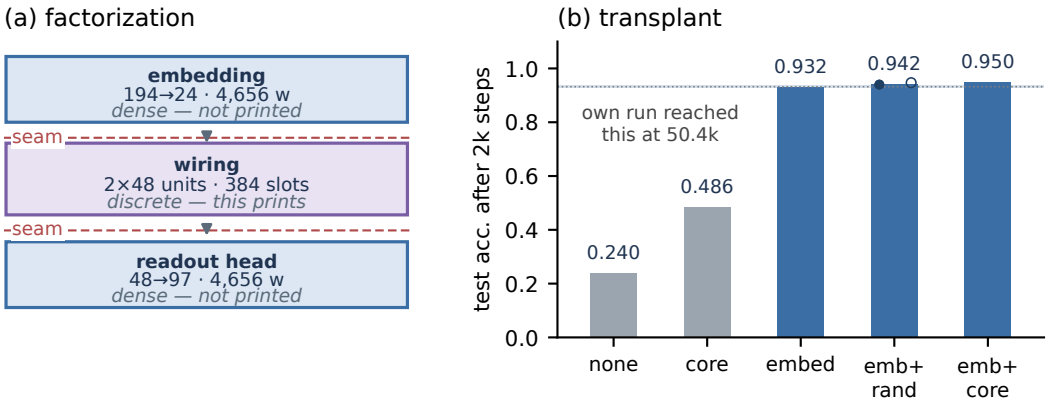


Figure 1: The substrate and the headline. (a) The factorization every argument in Section 3 is about: a dense embedding, discrete wiring, a dense readout, with parameter counts marked and the two seams the paper operates across drawn in. (b) Transplant conditions on the run's own 30k checkpoint after 2k fine-tune steps; the shaded band is B.2's $\pm\varepsilon$ measurement spread, and the embed+random bar carries three random draws, all 0.940, beside a non-critical-donor cell at 0.947. The host's own trajectory reached the embed-only level only at step 50.4k. An excerpt of the printed 193-line program, each wire annotated with its measured commitment step, is set as type in C.1.

matched statistic, $t_{0.9}$ 1.2k against the fresh-optimizer baseline's 12.7k (B.8). Its unaided trajectory reached 0.932 only at 50.4k, but that comparison carries the optimizer reset as well as the implant. Timing is always $t_{0.9}$, the first crossing of test accuracy 0.9 during fine-tuning. The schedule was compressed, not given a head start.

Reading raw checkpoints wire by wire, with zero training-time instrumentation, a bottom-up ladder of commitment times emerges — but which statistic names its bottom rung decides what the ladder means. Pooled spectral power is largely in place by step 1k, so it cannot be what the plateau grinds on. What the modular-addition battery measures is therefore the **exploitable** component: the one that, supplied, most accelerates a stuck host. That is not the **deferred** component, the one training spends the plateau building — the wiring, on both. The *films* of Section 4.1 *can* see what training builds: wiring forms, hardening follows, printability comes last (E.1). "Structure preceding the grok" is level-dependent.

Our contributions are three: a glass-box substrate that groks and prints faithfully; a causal dissociation that locates the exploitable component, with its decisive synthetic control; and a two-rung commitment ordering. The ledger gives each with its evidence and its limits — what we claim, what backs it, what we do not claim. *Standing.* **(1)** The substrate groks (17 of 19) and prints faithfully — four programs within 1.7 points of soft, 98.95% agreement, both fidelity confounds controlled (Section 2, A.1, A.6); fidelity, not improvement. **(2)** On modular addition the *exploitable* component is the embedding: $t_{0.9}$ 1.2k against the fresh-optimizer baseline's 12.7k, 16% of the parameters beating the other 84%, circuit identity worth a median 221 further steps on 8 of 8 hosts of one trajectory (Section 3.1, B.3, B.8); three independent lineages replicate the embedding arm, their baselines censored at 20k so those factors are bounds — as consistent with amplification as with supply (B.11). **(3)** The dissociation is not "dense maps are slow": the capacity-matched head crosses at 3.79k, 3.1–7.4× over three hosts, so *at least* three times slower; whether it is input-side or seam-side is open (Section 3.1, B.8). **(4)** On modular addition structure rescues where norm alone does not, per host rather than pooled (Section 3.2, B.5, B.9); deep parity hosts are the exception, where any full-scale implant un-sticks them (B.10). That half transfers to an MLP and a transformer (Section 3.4, B.9); we do not claim radius is *necessary*. **(5)** The sufficient class is cosines on a frequency set shared across both input positions (Section 3.2, B.11) — not frequency identity, not learned content. **(6)** Implantation helps while the host still lacks that component and stops once it has built one (Section 3.3, B.11); most of the post-grok deficit is transient repair, and what the component *is* the battery does not say. **(7)** The embedding is exploitable on both task families; the *wiring's* exploitability inverts under an identical implant — harmful for 2,500 steps and crossing at 6.2k–9.6k on modular addition, 1.1k–1.7k with no harmful phase on parity (Section 3.5, B.8, B.10).

What parity's embedding defers is its *allocation* of power to the support bits, not its convergence, which runs ahead of the wiring's throughout (D.5); we claim no inversion in convergence order. **(8)** A two-rung commitment ordering, 19 of 19, layer 1 unresolved at budget and ratcheting rather than locking (Section 4.1, D.3); at full resolution n = 3.

Withdrawn, with what survives each. **(9)** The five-arm ordering — the head arm spans 3,788–6,338 across hosts; the embedding's lead and head-beats-complement survive at all three (B.8). **(10)** The 14-of-19 last-change ladder — the statistic is undefined for a process still moving at cutoff; the commensurate 19-of-19 pair survives (Section 4.1, D.3). **(11)** Any law in lattice distance — a five-draw phase null spans 3,000–5,250; the 2.9k-against-9.75k contrast survives (Section 3.2, B.11). **(12)** The sharing rung as a timing statistic, and the registered test built on it — its 19-of-19 ordering survives, its timing does not (D.3, B.12). **(13)** The scoring of the isogenic post-grok control — nothing survives it, and the cross-run basis confound stays open (B.11, B.12). Frequency-competition dynamics are in Appendix D.

2 A Glass-Box Substrate that Groks

Every hidden unit computes the same fixed primitive, $\mathrm{eml}(x, y) = e^x - \ln y$, with two numerical safeguards given in A.5. It learns which four signals feed it — two per argument, through softmax choosers annealed at T = 0.5 — with a gain on each and a bias, 576 real coefficients across the network (B.4). Width 48, depth 2, embedding dimension 24, roughly 30k parameters; the task is modular addition at p = 97, one-hot pairs, a 50% train split, full-batch AdamW.

Figure 1a is the whole architecture. Fixing the primitive concentrates every learnable discrete choice into one finite object: 384 argument slots, each picking one of 25 signals at layer 0 or one of 73 at layer 1. That is about 2,080 bits in total, and that finiteness is what makes the network printable. The safeguards are the price of that unstable family (Trask et al., 2018). What the design buys is the *seam*: a dense input map, then discrete routing, then a dense readout, with the boundaries fixed by architecture rather than chosen by us. **Hardening** is the procedure that collapses the choosers to their argmax — a 30k-step anneal with a straight-through tail, defined once in A.5 and used unchanged everywhere.

Grokking is reliable here, and recognizable. Across 19 runs, 17 grok; the 2 others were censored mid-transition (Figure 4). Grok times — first step with train ≥ 0.99 and val ≥ 0.95 — span 28k–135.5k (median 85.5k), a 4.8$\times$ spread at fixed hyperparameters. Transition widths are 8.5k–48k (median 16k) under a convention frozen in Appendix A. The phenomenology matches prior accounts (Nanda et al., 2023; Gromov, 2023; He et al., 2026a): 71% of the embedding's oscillatory power (DC excluded) concentrates in 6 frequencies. The main run's total parameter norm rises to $\sim$151 by 27k and falls to $\sim$110 by 60k, with its embedding's own norm tracing a sharper arc ($\sim$79 at 20k to $\sim$38 by 80k); across the 19-run population, which excludes it (A.1), that 20k figure spans 48.1–74.0 (Liu et al., 2023).

Printing is faithful. All four grokked networks the recipe has been applied to harden into discrete programs within 1.7 points of their soft accuracy. The flagship prints at 98.36% against its teacher's 98.49%, a cost of 0.13 points, and agrees with the soft network on 98.95% of all 9,409 inputs. The claim of record is therefore fidelity rather than improvement (A.1). Both fidelity confounds are controlled: continuing *soft* training for the same +30k steps reaches only 97.58%, and the same recipe from random initialization yields 8%. The recipe can only print what training already built. The released artifact is the printed program itself, and the two dense maps do not hide the computation: the embedding is a wave bank, the readout a three-frequency matched-filter decoder (Appendix C).

3 Locating the Exploitable Component: The Transplant

3.1 The intervention and the dissociation

The donor is the flagship run at 100k steps, grokked. The host is the *same run's* 30k checkpoint, sitting mid-plateau at 0.203 test accuracy — climbing, not at floor. We copy one component from the future into the past and fine-tune everything for 2k steps.

Five conditions (Figure 1b). Against a no-implant control at 0.240, the donor's core circuit units alone reach 0.486 and the embedding alone **0.932** — 0.763 with the implant frozen (B.7). Adding the donor's core units reaches 0.950, randomly initialized replacements 0.940 over three draws, and six non-critical *donor* units 0.947 (B.1). So once the embedding is supplied, whose circuit occupies those positions barely registers. The core set is the six layer-0 units whose ablation — zeroing, throughout — costs more than $\delta = 0.05$ (B.1).

That 2k endpoint sits near ceiling, but the ±1k host-shift bootstrap is a *paired* design and reads the other way. B.2's core-augmented arm — donor core logits at both layers, a superset of the layer-0-only condition above — beats embed-only at all three hosts. The margins are 0.021, 0.025 and 0.027 against a 0.9-point within-condition spread. Across eight plateau hosts the donor's core — the same two-layer arm — crosses 0.9 ahead of randomly initialized replacements every time, by a median 221 steps — 17–58% of the random-core arm's crossing time. Those eight are checkpoints of one trajectory, so the lead is consistent across host maturity rather than across independent runs (B.3). Circuit identity is therefore worth something, and what it is worth is about 1.9% of the $\sim$11.5k steps the embedding buys.

The symmetric battery, and how much of it survives a shifted host. The strongest objection is the missing converse: implant everything *except* the embedding. Every coherent component eventually rescues over a 20k horizon, ordering embedding 1.2k, head 3.79k, complement 4.9k, wiring 9.6k, baseline 12.7k on the seed-0 host (Figure 2d). Two relations hold at all three hosts: the embedding is fastest by a wide margin, and the capacity-matched head beats the complement that strictly contains it. The five-arm ordering does not — the head alone spans 3,788–6,338 (B.8). A third relation is narrower than we first read it. The wiring arm is donor routing over the host's own gains, not a finished circuit. It spends its first 2,500 steps *below* the untouched baseline at the 30k host, and is still below it at the 2k read at 29k. At 31k it is already above (0.438 against 0.344), so the early harmful phase belongs to the younger hosts rather than to the arm (B.8).

One arm of that battery answers the accounting objection by measurement. The readout head carries *exactly* the embedding's 4,656 weights (plus 97 biases; B.4) — and is a dense spectral object of the same kind (C.4). Implanted alone — same capacity, same spectral character, opposite end of the computation — it is harmful at first like the wiring arm. Across the three hosts the ratio reads 3.1×, 3.3× and 7.4×, so **at least three times slower** is what the measurements support, not exactly three. The dissociation is therefore not "dense maps are slow". Whether it is specific to the input-side *representation* is less settled: the embedding's input basis is fixed, while the head's columns index host units whose meaning differs from the donor's, and only the embedding is read by routing that re-selects cheaply (B.7). The asymmetry may be about which side of the seam a component sits on, and the MLP's absent ordering fits that reading too (Section 3.4).

3.2 Controls: structure, not norms

The radial reading of (Liu et al., 2023) — that the transplant merely moves the host to the right radius — is tested on two cross-run hosts (Figure 2a; Appendix B.5). On both, structured implants rescue (0.659–0.961) while norm-matched controls do not: rescaled host structure lands within 0.014 of baseline and random directions at or well below it. Both controls fail on the mature flagship host too, where there is real partial structure to amplify: rescaled host structure buys an early boost (0.533 at 2k) and then plateaus at 0.905 by 20k. That is 6.3 points *below* the untouched baseline's 0.968 at the same horizon (B.9).

The rescue is frequency-agnostic too: another run's frequency family rescues, under a donor-norm confound that leaves the own-versus-other ordering unread (B.5). The conclusion *refines* the radial account rather than rebutting it: radius alone never suffices here, while class membership at a workable scale does, per the next control. The fastest implant here is Section 3.1's donor embedding carried over unrescaled, norm 43.66 into a host at 74.13, while on parity it is the host-matched arm that is fastest (B.10, B.11). Both radii work, so radius is not the governing variable.

The synthetic-embedding control. An untrained cosine bank — six frequencies, four dimensions each, random phases per position, scaled to donor norm — rescues the 30k host at 0.844 by 2k; six *randomly chosen* frequencies give 0.828. Which property is load-bearing is sharp (Figure 2b; B.11): rank-6, smooth-aperiodic and periodic-but-position-*unshared* implants all track the fresh-optimizer baseline, while shared cosine banks on the lattice cross by 2.9–3.9k. The sufficient class is cosines

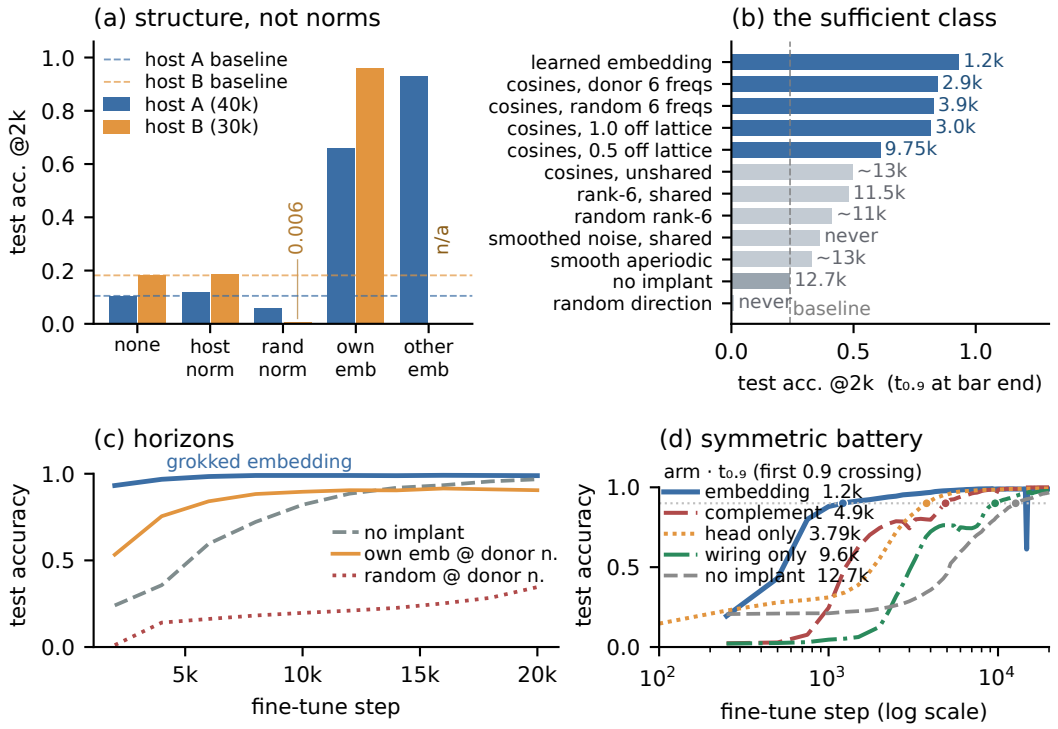


Figure 2: The transplant battery: structure rescues, norms do not. (a) Host-normed controls sit within 0.014 of their hosts' baselines and random-normed controls at or well below them, while the three wave-structured implants rescue at 0.659–0.961 (B.5). (b) The class decomposition (Section 3.2, B.11): bar length is accuracy at 2k, the label at each bar end is $t_{0.9}$. Position-shared cosine banks rescue; the registered shared-but-aperiodic cells, and the low-rank, unshared and random-direction implants, all track the fresh-optimizer baseline. (c) Matched horizons on the seed-0 host over 20k steps (B.9); its axis starts at 2k, so the early sub-baseline dip every arm shows appears only in (d). (d) The symmetric battery on a log axis, same host; each arm's first 0.9 crossing is marked on its curve and given in the legend, and Section 3 withdraws the five-arm ordering as host-dependent (B.8). The embedding arm's single-eval dip near 15k is real, left unsmoothed.

on a frequency set *shared across the two input positions*, at or near the task's characters (Chughtai et al., 2023). A rank- and frequency-count-matched sweep puts the whole effect in a step between zero and *two* shared frequencies, with no further gain beyond it: sharing is necessary without being a dose (B.11).

Sweeping the offset from that lattice yields one readable contrast and no more: a peak of 9.75k at the half-integer against 2.9k on the lattice. A five-draw phase null on the on-lattice cell alone spans 3,000–5,250, which covers all but one of the intermediate offsets, so we **withdraw any reading on which distance from the lattice governs the rescue** (B.11). A full-integer offset returns at 3.0k, but two of its six frequencies wrap onto donor characters, so we do not rest a periodicity claim on it either. Within the class the learned embedding retains a $\sim$2.4$\times$ advantage on the draw of record (1.2k against 2.9k), 2.5–4.4$\times$ across the phase draws.

3.3 Which hosts and donors, and when

The rescue is not a property of one pair. All three grokked donors tried against the seed-0 host beat baseline, and two rescue fully (0.932, 0.969, 0.499). At modulus 113, a 10k host reaches 0.924 with the implant against 0.374 without and 0.227 with the norm control (B.5). An 8-pair cross-run battery then complicates the tidy version. Above a donor-consolidation floor, nothing about the donor predicts rescue quality; host-side variation dominates. The 0.499 donor rescues a *different* host normally, so it is a pair effect rather than a bad donor.

When matters more than *who*. Implanting into the flagship's own hosts from 10k to 40k gives a flat-high 0.910–0.941 against fine-tune-only controls climbing 0.135 $\rightarrow$ 0.828 (Figure B1b). The cross-run battery shows that series was measuring host *age* where the real variable is *phase*. Deep-plateau hosts rescue (seven, median 0.69, range 0.480–0.971). Both mid-transition hosts are helped substantially, so the registered no-gap prediction failed 2 of 2. Hosts that have essentially solved the task are never helped, and at the 2,000-step read they sit at or below baseline in three of three, the largest deficit 0.632. Most of that deficit is repair rather than damage: the implant resets the host to chance first, and read to 12,000 steps every cell but one returns to its own baseline (B.11).

The boundary is not the grok criterion but whether the host has effectively solved the task: i0_d1 at 0.913, six thousand steps *before* it formally groks, already shows the deficit (-0.160 at 2k). Implantation helps while the host is still building its solution, and stops once it has one. *What* the implant supplies the battery does not say (B.11).

3.4 Other substrates

Half the result transfers to a plain MLP, and precisely which half. The grokked first layer rescues a memorizing host (0.862 against 0.440) while norm controls stay dead, but the complement rescues at least as well (0.990 against 0.862). There is therefore no rank ordering. An MLP has no architectural seam — its later layers are themselves representation-bearing — so it cannot separate rebuild times as the EML factorization can. A one-layer transformer (Nanda-style recipe; groks at $\sim$16.75k) carries the structure-versus-norm half in *graded* form; we did not run its complement arm, so it does not speak to component ordering. The grokked token embedding rescues its memorizing checkpoint within $\sim$200 steps, while a random direction at the same norm is destructive early and $\sim 20\times$ slower. But that norm control does eventually un-stick this host, unlike the EML and MLP cases — an Omnigrok-consistent radial contribution under a $10\times$ larger weight decay (B.9). Circuits-are-cheap is claimed only in the glass box, where it is established.

3.5 A second task family: sparse parity

(40,3)-sparse parity (Barak et al., 2022) groks here with textbook phenomenology — tens of thousands of steps at chance over a perfectly memorized training set, then a snap to 0.998 — but only in some runs. Sixteen lineages have been trained at identical hyperparameters. **Five** grok after a plateau (plateau lengths 60.5k–116.5k for the four filmed; a fifth groks at 44.5k), **eight** generalize with no plateau at all (grok times 0.5k–11.5k), and **three** never grok inside the budget. The films and the battery below use the first eight; the later eight strengthen D.5's negative control and are reported there.

The films cover all four plateau lineages, and the inversion replicates on each, as does the battery. Donor wiring rescues at $t_{0.9}$ of 1.7k, 1.1k, 1.35k and 1.20k, against baselines of 6.9k, 5.6k, 8.85k and — on the fourth — a host that never reaches 0.9 unaided. The donor is taken at grok on three of the four and at the final checkpoint on the first, whose cleanup happened to preserve half its solution circuit (B.10). The inversion is a claim about runs that plateau, not about the task.

Yet the films locate parity's hidden progress in the layer-0 *wiring* (Figure 3b), and the battery does not contradict them so much as speak to a different question. It is true that the embedding arm is fastest or tied on parity as on modadd: exploitability is task-independent. But the *wiring* arm behaves differently across the tasks under an identical implant. On modular addition it is harmful over its first 2,500 steps and crosses at 6.2k–9.6k across the three hosts (Section 3.1); on parity it crosses at 1.1k–1.7k against the four baselines listed above, with no harmful phase on any lineage. We rest on that contrast, not the five-arm ordering, which is host-dependent (B.8).

The embedding never *amplifies* the support bits until cleanup. Its relevant-bit power sits at chance on both trajectories through the plateau proper and has reached only $\sim 2\times$ chance by the snap, against the wiring's move from 0.281 to 1.000 over the same window. A rotation-invariant linear probe sharpens rather than overturns this, finding them at chance on one trajectory and merely in the upper tail of the irrelevant distribution on the other (D.5). What training defers is the allocation, not necessarily the information. Layer-0 wiring meanwhile converges toward its at-grok program across the approach on both runs.

The surgery closes the loop on the ripe (already grokked) host. On modadd the wiring was the weakest arm and initially harmful; here it supplies most of the wait. It replicates on the second trajectory under a condition the first lineage hid: **the donor circuit must be taken at grok.** Cleanup can rewire layer-0 away from the program that solved the task; the second run retains almost none of its at-grok wiring by 150k. A naive final-checkpoint implant therefore carries a circuit that no longer works and finishes *below* baseline. That is the circuit-side twin of the embedding scale-matching law, it is measurable on the donor with zero training, and both stages are reported (B.10).

4 The Ladder and the Ratchet: When Structure Commits

4.1 The margin film — and what "lock-in" turns out to mean

Because the wiring is a set of choosers, every raw checkpoint records each wire's current choice and how strongly it is held. We call that record a **film**: a per-wire diary of argmax and margin, read from saved checkpoints with zero training-time instrumentation. A wire *locks in* at the last checkpoint after which its choice never changes again.

Ordered by lock-in time, the components form a bottom-up ladder — embedding spectra, then core layer-0 wiring, then the rest of layer 0, then layer 1 — but the middle two rungs are tied in two of three films (D.1), so the two-rung claim below is the one we make.

The bottom rung is early, and we checked that it is not early by construction. The spectra statistic is a threshold crossing where the wiring rungs are last-change times, so we recomputed layer-0 as a threshold crossing too, halfway from its initial to its final match. Under that commensurate pair the ordering holds in **19 of 19** (D.3). Under that same recomputation the spectra rung lands at 500–900 steps in 17 of 19 runs (D.3 keeps the diary convention's 0.5k–1.3k separately). The top rung is not resolved at all. Layer-1 last-change medians run from 132.5k to 147.1k against a 150k budget, so layer 1 is still moving at cutoff in **every** run, not only in the five we previously labelled saturated. A last-change statistic is not well defined for a process that never stops changing: its expectation simply tracks the budget. Reporting "layer-0 median $\leq$ layer-1 median" as a population result in 14 of 19 runs overstates what a censored statistic can carry, and we withdraw that framing.

What survives is a two-rung ordering with an open top: embedding spectra settle very early, and core layer-0 wiring resolves at a run-dependent time tens of thousands of steps later. At full resolution that ordering is an n = 3 statement; the commensurate population result above is the one we report, and D.3 keeps the last-change figures and their cohort split for continuity. Extending three runs to 400k–500k, so that layer 1 can resolve, is the experiment this section needs and has not had.

Lock-in is not finality, and the distinction remakes the top rung. A wire can settle confidently on a choice hardening later overrules, so "never changes again" and "matches what finally prints" are different events. Layer 0 behaves as the ladder suggests, touching a perfect match to its hardened program at 98k. Layer 1 does not lock at all: it *ratchets*, correct selections persisting at about 0.97 per 2k steps while the others churn at 0.46–0.89. No discrete late event appears anywhere, and hardening supplies the last third (D.3). That is a better description of cleanup than a jump (Nanda et al., 2023). It holds on parity too, on all five lineages *including the one that never groks*, so the ratchet is architectural rather than a signature of generalization (D.5).

Section 3 suggests why the ladder has this shape: everything downstream of the representation is cheap, so the slow rungs wait on the embedding. That link, though, is an inference across two measurements rather than a measurement itself, and the battery speaks to what a host can be handed rather than to what the ladder is climbing. "Structure precedes the jump" must always name its level *and* its task.

5 Related Work

Five accounts bear directly on the claim; the rest are in Appendix G.

The representational thesis, and its causal upgrade. (Liu et al., 2022) made the thesis explicit and (Liu et al., 2023) gave it a radial mechanism. Ours is the causal version: we implant learned structure into a live trajectory — including its own — with norm controls that test the radial account directly

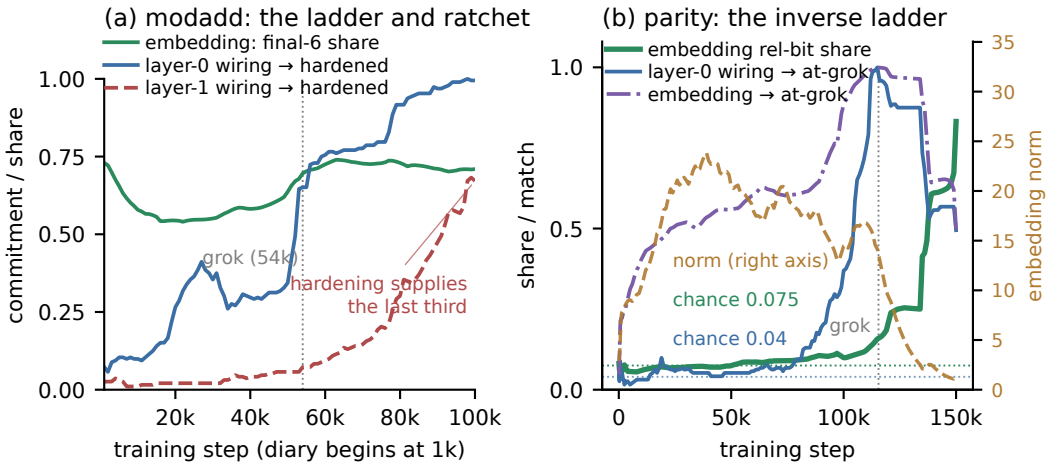


Figure 3: The ladder, the ratchet, and the inversion. (a) Modadd, from raw checkpoints (diary from 1k). The six frequencies are taken from the 100k checkpoint and tracked *backwards*, a construction that inflates early values: read it as how much power the eventual winners already hold. Layer-0 wiring touches 1.000 match-to-hardened at 98k while layer-1 accretes to 0.672, hardening supplying the last third; layer 1 is an unresolved rung rather than a ladder step (Section 4.1). Dotted vertical: grok at 54k. (b) Sparse parity, same filming. The dash-dotted **embedding-to-at-grok** trace, the matched counterpart of the wiring's, rises throughout and sits above it at every step, and the wiring's chance floor of **0.04** is also where its 0.05–0.09 plateau band sits. The claim is deferred *allocation* of support-bit power, not an idle embedding — the inverse ladder (Section 3.5, D.5).

(Section 3.2). (AlquBoj et al., 2025) reaches embedding-centrality from the opposite direction, showing that *inserting* an embedding into an MLP induces the delay while embedding-free MLPs generalize immediately, which they attribute to a bilinear embedding–downstream coupling. Their result says the embedding *creates* the delay; ours measures what the delay is *spent on*, and their coupling is the natural theory for the rebuild-time asymmetry we report.

Interventions that act before or during training. (Minegishi et al., 2025), (Xu et al., 2025), (Howe, 2026a) and (Howe, 2026b) all act at or before the plateau; Appendix G places each against our controls. With those added: a mature representation given at birth prevents the plateau, and given mid-plateau ends it.

Component-resolved interventions elsewhere. (Gomezjurado Gonzalez, 2026) dissociates an encoder from a decoder on Collatz prediction and reaches the *opposite* localization, reading its delay as limited access to structure the encoder already holds. Being an initialization-time intervention with a fresh recipient, it bounds our novelty claim rather than competing with it.

The nearest surgical and decompositional relatives. (Truong, 2026a) builds cross-trajectory chimeras splitting weights into norm and direction, reading *which* solution a recipient reaches, across independently seeded runs. We instead read *how long* a host takes to rebuild what an implant leaves missing, and our host and donor are one run at two times. Their Appendix-K init-projection reading does not transfer here ($p = 0.29$, F.4). (Truong & Phan, 2026) separates detectability, transplantability and causal persistence across independently trained networks and establishes only the first; we test the other two within one run. (Chou et al., 2026) decomposes grokking into slow representation learning and fast readout calibration, reporting that ordering on (40,3)-sparse parity, our own setting, where we localize the wait in the wiring. The readings agree at different resolutions rather than conflicting. Their encoder is our embedding *and* our wiring together, so a two-way split cannot ask which of the two is slow. Their gradual pre-jump representation improvement is, in our factorization, the layer-0 wiring climbing while the embedding withholds *power* from the support bits.

Group structure. (Chughtai et al., 2023) show small networks learn a group's irreducible representations — for modular addition, the Fourier characters. Section 3.2 puts that in causal form: implanting cosines on a frequency set shared across both input positions ends the plateau, fastest on

the character lattice itself (B.11). We claim that contrast and no law in lattice distance, but it puts the group's structure in the intervention and not only in the trained weights.

6 Discussion and Limitations

Is the inversion a fact about grokking, or about our factorization? Modular addition's features are frequencies, which live in the embedding; parity's are the identities of three bits, which the wiring can express. So the inversion might reduce to "feature identification is slow, and this architecture stores it in a different module per task."

Three measurements say the localization is discovered rather than assumed. First, nothing prevents parity's embedding from carrying support-bit structure: a dense map over 40 inputs can amplify three coordinates. Yet on all four plateau lineages its relevant-bit power stays near chance until after the snap, while the layer-0 wiring climbs steeply across the approach. Second, the components are defined once, by architecture, and the same probes apply unchanged to both tasks. Third, the sufficiency side is task-independent — any coherent finished component accelerates a primed host — so the inversion is carried entirely by training-side measurements, which could have agreed across tasks and did not. The objection does correctly predict our scoping: where there is no seam, as in the MLP, the battery cannot separate components.

A radial clock leaves the timing unexplained. (Liu et al., 2023)'s t ˜ $1/\gamma$ is a radial clock, since sharpened into quantitative delay laws (Truong et al., 2026; Truong, 2026c). Our 19 runs hold γ fixed and still span grok times of 28k–135.5k: a radial law fixes the clock's scale, not the seed-dependent residual within it. Everything measured here is angular structure that radial theory integrates out. They de-grok by constraining radius; we accelerate by implanting angular structure — and radius *without* structure is actively harmful (B.9).

Why this matters beyond grokking. Our own second task forbids concluding that representations rather than circuits are the natural object to audit: on parity the routing is what the plateau builds. *Which* component repays watching is therefore a property of the task–architecture pair, and not a general fact about representations. The donor's component *inside* the host's own row space is better early than the part outside it, and crosses no later on two hosts of three (B.11), so the host does not simply lack a subspace — and rescaling the host's *own* structure to donor norm ends below the untouched baseline at 20k on every host (Section 3.2), so this is not amplification of host content either (Varma et al., 2023). The control establishes less than the question it was built for: our supply reading is refused, amplification is not established, and the whole donor embedding is about three times faster than either half (B.11). What generalizes is the method — ask which component a stuck host can exploit, measure it against what training defers, and report the control that refuses your framing.

Limitations. All results are small-scale, on two task families at two moduli, and on one *instance* of each — every parity lineage shares the same three support bits. Several load-bearing cells are single measurements on a single host: the synthetic decomposition and its lattice sweep, modulus 113, the MLP and the transformer. B.2's host-shift bootstrap and B.3's eight-host sweep cover the headline arms, not these.

Parity's inversion is measured on the four of the first eight lineages that plateau and then grok — a fifth, i8, plateaus and never does — and is retrospective, not predictive (D.5). Donor generality rests on 12 cross-run pairs with outcomes spanning 0.355–0.971 and no predictor above the consolidation floor (B.11). The circuit dissociation is shown only where the seam is architectural, and the MLP and transformer transfers cover structure-versus-norm alone: the headline is a fact about modular arithmetic, generalized only as "transplant the exploitable component". The wiring's ≈2,080-bit hypothesis space is far smaller than the dense maps' capacity, and B.4 says why the dissociation is informative anyway. The printed program is a translation at 98.95% agreement whose epsilon every anatomical claim inherits. Finally, everything here concerns clean labels.

Reproducibility Statement

Artifact. Code, the 19 dense checkpoint diaries, the flagship's 193-line printed program, the run logs for every battery that wrote one, and the timestamped research log are provided as supplemen-

tary material with this submission. The other three prints of C.2 were scored at the time and their accuracies are in the log, but their program text was not archived and is not included. Two batteries likewise wrote no machine-readable log and are reproducible only from the released code plus the research log's transcribed results — the eight-pair cross-run battery and the original synthetic-bank sweeps behind B.11's sufficiency table, both of which printed to a terminal rather than to a file. We name them rather than leave the coverage claim unqualified. The log carries the pre-registration and outcome for every battery in this paper with two exceptions we flag rather than paper over. The first is kill #21, reported in B.12, has a dated contemporaneous entry recording its outcome but no registration written before that outcome was known — the cross-host run it reports was queued as a consistency check rather than against a stated threshold. We did not reconstruct a registration for it after the fact and backdate one, because a ledger whose entries may have been written after their outcomes is worth less than one with an acknowledged gap. The second is a lapse in the log's *narration*: it runs contemporaneously to 2026-09-14 and then stops, while the work continued through kill #42. The registrations did not stop — from #39 onward each prediction and its falsification threshold was written as the docstring of the script that runs it, and that script was committed to version control before the run. The released log therefore carries one block, dated the day it was written and labelled a reconstruction, that narrates #39–#42 and the controls around them from those committed docstrings and the run files' timestamps; nothing in it is backdated, and `runs/registration_timestamps.txt` gives the commit-versus-outcome times for every one of them, including the rows that do not help us.

Determinism. All training and every fine-tuning battery are full-batch, so runs are deterministic given a seed, a checkpoint and a fixed numerical configuration: every number in this paper reproduces exactly rather than in distribution when re-run under the configuration that produced it. It is not invariant to that configuration. Changing only the BLAS thread count leaves settled hosts alone (shifts of 0.0007 and 0.0015 on two of B.11's post-grok cells) and moves a boundary-phase cell by 5.4 points, which is six times the measurement epsilon B.2 quotes from the host-shift bootstrap; B.2 states the envelope and the released artifact records the thread count per script. We report this because we found it by failing to reproduce one of our own numbers, and the number that was right was the published one. The population is 19 runs: one initialization seed crossed with ten data seeds (i0_d0–i0_d9) plus nine further initialization seeds at one data seed (i1_d0–i9_d0), under the naming of A.1 — the cohort split D.3's ladder caveat turns on; the transplant batteries name their host checkpoint, donor checkpoint and draw seeds in place. Fine-tuning uses a freshly constructed AdamW at the training settings of A.5 (lr 10^{-3}, weight decay 0.1) — the optimizer state is *not* inherited from the host. Every arm, implant and baseline alike, therefore carries the reset, so it cannot produce the between-arm gaps; what it contributes on its own is not measured, since no arm inherits the host's optimizer state.

Compute. Everything runs on a laptop CPU except the transformer battery, which uses one consumer GPU. The modular-addition population is 19 runs $\times$ 150k full-batch steps $\approx$ 2.85M optimizer steps. The rest of the catalogue is larger than the population: 16 sparse-parity lineages, twelve of them to the full budget $\approx$ 1.8M steps; the 32-run MLP census behind F.1 and F.5, $\approx$ 4.8M; the distillation probe of E.1, 11 teacher frames $\times$ 6 student restarts $\times$ 10k $\approx$ 0.7M; and the transplant, norm, synthetic-class, horizon, parity and dose-response batteries, $\approx$ 0.5M. The aggregate is on the order of 10M optimizer steps. Measured throughput on the CPU used here is $\sim$28 full-batch steps per second at this model size (the eight-host B.3 battery, 80k steps, completed in 47 minutes), which puts the EML population on the order of 30 CPU-hours and the whole catalogue on the order of 100; the MLP and parity models differ in size and their rates were not separately measured. These are measured-rate estimates, not billed figures.

Pre-registration. Every battery cell was run against a prediction recorded before the outcome was known, with the falsification threshold fixed alongside it: in the research log through kill #38, and from #39 onward in the committed docstring of the script that runs it. The log is released with its failures intact, including every entry where a prediction was falsified; Appendix B.12 indexes them, and `runs/registration_timestamps.txt` dates each registration against its outcome. Three edits have touched the file, all visible in the released code. Two are appends: the reconstruction block, then a later contemporaneous entry after it, so the block is not the final entry. The third is an annotation inside that block, flagging an unsupported leg of the isogenic entry and adding the measurement that verifies two others. A fourth pass redacts five references to submission logistics — a venue name, two registration lines and three commit hashes — each marked in place as a redac-

tion; no entry, prediction, outcome or date is affected. Nothing else was removed and nothing was backdated.

ETHICS STATEMENT

This work trains small networks from scratch on synthetically generated arithmetic and parity tasks. It uses no human subjects, no personal or scraped data, no human annotation, and no pretrained model. We see no dual-use or deployment risk arising directly from it. The one claim with a bearing outside the paper is Section 6's: that which component repays auditing is a property of the task and architecture rather than a general fact, since on parity it is the routing and not the representation that training spends the plateau building. Section 6 declines the tempting general form of this — that representations rather than circuits are the natural object to audit — and we repeat that refusal here, because the general form is the one an auditor would be most likely to act on. Whether even the task-dependent claim survives scale is open, and we would not want it relied on as a safety argument at scales where it has not been tested.

LLM/AI USE DISCLOSURE

Large language models were used substantially in the preparation of this work, and the authors take full responsibility for all of it — every claim, number, citation and line of code — regardless of how it was produced. The division is this: LLMs assisted in building the experiments and in drafting and writing the paper; the research questions, the hypotheses and the interpretation of what the results mean are the authors'.

Where they were used. (i) *Code.* Most experiment scripts, analysis code and figure-generation code were drafted with LLM assistance and then read, run and checked by the authors. What each experiment was for, and what its outcome meant, was decided by the authors. (ii) *Manuscript.* LLM assistance was used for editing, restructuring, compression to the page limit, and drafting appendix prose from the research log. (iii) *Verification and criticism.* LLMs were used adversarially against our own work: to audit numbers in the manuscript against the raw run logs, to search for internal inconsistencies, and to simulate reviewer criticism of successive drafts. Several errors reported in this paper's errata and several of the ledger's kills were first surfaced that way. (iv) *Literature.* LLM-assisted search was used to locate related work; every cited claim was then checked against the cited paper by the authors, and at least one LLM-supplied characterization of a related paper was found wrong and corrected.

Where they were not used, and the three exceptions. Every prediction in Appendix B.12 — the registrations behind every battery reported in this paper — was written by the authors and timestamped before the corresponding run was launched, in the research log through kill #38 and from #39 onward in the script docstring committed before the run, as the Reproducibility statement sets out. One exception sits *inside* this paper, and we name it here rather than leave a reader to find it. The cross-run replication of the headline timing reported in B.3 was registered in an **LLM-authored** docstring — its three predictions, their falsification thresholds, the censoring rule and the lineage-selection criteria — committed to version control before the run, with the runs, the data and the scoring as described there. The authors read that registration before launch and stand behind it, but did not write it; that one result therefore does rest on an LLM-authored hypothesis and threshold, and a reader who discounts LLM-authored registrations should discount it and fall back on B.3's own standing caveat, which it does not remove. Every other result in this paper rests on an author-written registration. Beyond it, the released log holds two further exceptions, both in the companion-program work whose ledger entries B.12 already places outside this paper: a registered prediction with its own falsification threshold that was LLM-authored, made on 2026-09-05 against the companion program's press run and scored **falsified**; and the design of the first and third Collatz variants, the first of which failed by a design flaw caught the same day. We name them here rather than leave a reader to find them in the log. No number in this paper was produced or transcribed by an LLM without being checked against the raw log or checkpoint it came from; where such checks changed a number, the change is recorded as an erratum in the released log.

Known risk. LLM assistance makes fluent, confident errors cheap to produce. Our mitigation was mechanical rather than stylistic: a single source-of-truth research log, values-of-record tables

that map every quoted number to its origin (A.6), and repeated adversarial audits of the manuscript against raw data. We report this because the mitigation is part of the method, not because we regard the risk as eliminated.

REFERENCES

H. V. AlquBoj, Hilal AlQuabeh, Velibor Bojkovic, Munachiso Nwadike, and Kentaro Inui. Mechanistic insights into grokking from the embedding layer. *arXiv preprint arXiv:2505.15624*, 2025.

Yamini Bansal, Preetum Nakkiran, and Boaz Barak. Revisiting model stitching to compare neural representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. arXiv:2106.07682.

Boaz Barak, Benjamin L. Edelman, Surbhi Goel, Sham Kakade, Eran Malach, and Cyril Zhang. Hidden progress in deep learning: SGD learns parities near the computational limit. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2207.08799.

Chi-Ning Chou, Oscar Uzdelewicz, Neng-Chun Chiu, Yao-Yuan Yang, and SueYeon Chung. Two speeds of learning: A representation-readout decomposition of grokking and double descent. *arXiv preprint arXiv:2605.27078*, 2026.

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 (ICML)*, 2023. arXiv:2302.03025.

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 (ICML)*, 2020. arXiv:1912.05671.

Laura Gomezjurado Gonzalez. The long delay to arithmetic generalization: When learned representations outrun behavior. *arXiv preprint arXiv:2604.13082*, 2026.

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*, 2026a.

Kaiyu He, Mian Zhang, Peilin Wu, Xinya Du, and Zhiyu Chen. Is grokking worthwhile? functional analysis and transferability of generalization circuits in transformers. *arXiv preprint arXiv:2601.09049*, 2026b.

Gunner Levi Howe. Structure-specific representational priors causally control the grokking delay. *arXiv preprint arXiv:2607.04333*, 2026a.

Gunner Levi Howe. What makes a representational prior work? feature families, label-free invariances, and critical windows in grokking. *arXiv preprint arXiv:2607.12735*, 2026b.

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 (ICLR)*, 2024. arXiv:2310.06110.

Jaerin Lee, Bong Gyun Kang, Kihoon Kim, and Kyoung Mu Lee. Grokfast: Accelerated grokking by amplifying slow gradients. *arXiv preprint arXiv:2405.20233*, 2024.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. arXiv:1411.5908.

Ziming Liu, Ouail Kitouni, Niklas Nolte, Eric J. Michaud, Max Tegmark, and Mike Williams. Towards understanding grokking: An effective theory of representation learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2205.10343.

Ziming Liu, Eric J. Michaud, and Max Tegmark. Omnigrok: Grokking beyond algorithmic data. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.01117.

Eric J. Michaud, Isaac Liao, Vedang Lad, Ziming Liu, Anish Mudide, Chloe Loughridge, Zifan Carl Guo, Tara Rezaei Kheirkhah, Mateja Vukelić, and Max Tegmark. Opening the AI black box: Program synthesis via mechanistic interpretability. *arXiv preprint arXiv:2402.05110*, 2024. Also published as "Opening the AI Black Box: Distilling Machine-Learned Algorithms into Code", Entropy 26(12):1046, 2024.

Gouki Minegishi, Yusuke Iwasawa, and Yutaka Matsuo. Bridging lottery ticket and grokking: Understanding grokking from inner structure of networks. *Transactions on Machine Learning Research (TMLR)*, 2025. arXiv:2310.19470.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2301.05217.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. In *ICLR Workshop on the Role of Mathematical Reasoning in General Artificial Intelligence*, 2022. arXiv:2201.02177; workshop held 2021.

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

Anand Swaroop. Latent algorithmic structure precedes grokking: A mechanistic study of ReLU MLPs on modular arithmetic. *arXiv preprint arXiv:2603.23784*, 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.

Andrew Trask, Felix Hill, Scott Reed, Jack Rae, Chris Dyer, and Phil Blunsom. Neural arithmetic logic units. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. arXiv:1808.00508.

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

Xuan Khanh Truong. At-grok is not converged: A measurement-validity audit for grokking representation metrics. *arXiv preprint arXiv:2607.06639*, 2026b.

Xuan Khanh Truong. What does the weight norm control in grokking? logit-scale mediation under cross-entropy. *arXiv preprint arXiv:2606.18465*, 2026c.

Xuan Khanh Truong and Thanh Duc Phan. Beyond coordinate gauge: An audited protocol for detecting donor-specific functional fingerprints after neural collapse. *arXiv preprint arXiv:2607.11967*, 2026.

Xuan Khanh Truong, Hoang Viet Doan, Duc Trung Luu, and Thanh Duc Phan. The weight norm sets the grokking timescale: A causal delay law. *arXiv preprint arXiv:2606.13753*, 2026.

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

Zhiwei Xu, Zhiyu Ni, Yixin Wang, and Wei Hu. Let me grok for you: Accelerating grokking via embedding transfer from a weaker model. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2504.13292.

Shakir Yousefi, Andreas Plesner, Till Aczel, and Roger Wattenhofer. Mind the gap: Removing the discretization gap in differentiable logic gate networks. In *Advances in Neural Information Processing Systems*, 2025. arXiv:2506.07500.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. The clock and the pizza: Two stories in mechanistic explanation of neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2306.17844.

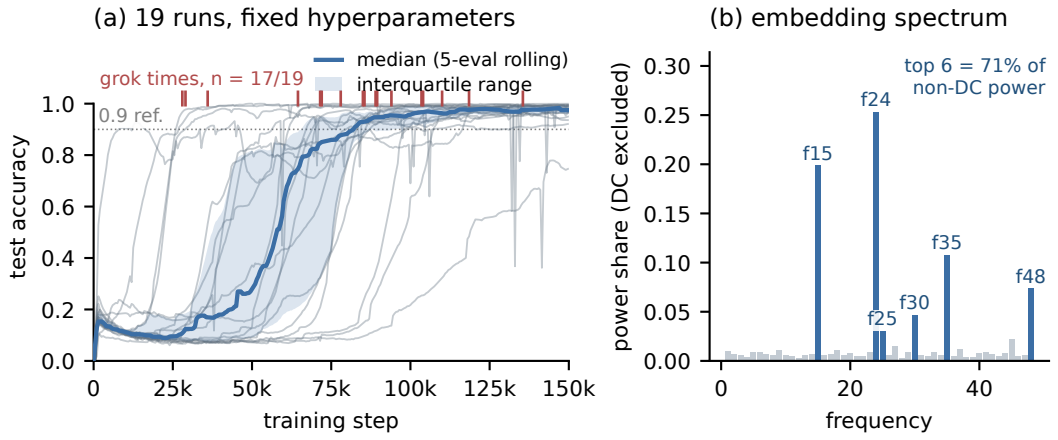


Figure 4: (Appendix A) — The substrate groks, reliably and recognizably. (a) Test accuracy for all 19 runs (17 grok; times 28k–135.5k, median 85.5k; widths 8.5k–48k, median 16k, under Section 2's frozen convention), each run's grok time ticked along the top, the heavy line the across-run median under a 5-eval rolling median, the band its interquartile range and the dotted line 0.9 for reference. Censored runs and the taxonomy: Appendix A. (b) Main-run embedding power spectrum, DC excluded, pooled over both input blocks: 71% of power in the six labelled frequencies. That the same six are *shared across positions* is A.6's per-block reading (69%, 73%), which a pooled spectrum cannot display.

Appendices do not count toward the 9-page limit. Every number below is a value of record; two per-run tables carry explicit reproduction pointers in place of unrecoverable cells.

A Numbers of record, timing table, and hyperparameters

A.1 The seed-mapping timing table

Every timing range quoted in the main text is checkable against the table below, whose every cell is computed from the released diaries under the frozen conventions. Columns are labeled with their statistic: **t50** is the first step at which *test* accuracy crosses 0.5; **grok time** is the first step at which train accuracy reaches 0.99 *and* validation accuracy reaches 0.95 (first-crossing, no holding clause; Section 2); **width** is the 10–90% interval of the final rise under the frozen convention of A.2; **class** is the four-phase taxonomy of (Liu et al., 2022) (A.3). The population ranges of record, over the 19 population runs (the three early film runs are listed but excluded from population statistics): grok times **28k–135.5k, median 85.5k**, over the 17 grokked runs (i0_d3 at 28,000; i1_d0 at 135,500 — a 4.8× spread at fixed hyperparameters); widths **8.5k–48k, median 16k**, over the same 17. t50 first-crossings span **1.5k–119k** over all 19 — the 1.5k is i4_d0's transient early crossing (footnote A.4) and the 119k is the *censored* run i9_d0's crossing, which is itself why the population claim rides on the grok-time statistic; over the 17 grokked runs, excluding i4_d0's transient, t50 spans **13.5k–98k** (i6_d0 to i0_d6). The best run is i6_d0: it groks at 29k and reaches test accuracy 1.000 at 49k, the substrate's first perfect score.

Run	t50 (test-0.5)	Grok time	Width	Final acc (mean of last 5 evals)	Class
seed-0 flagship (wd0.1_cpu)[1]	35.4k	54k	13.6k	0.985 (0.9849 last-eval; 0.9836 hard-ened)	grokking
seed 1 (wd0.1_s1)[1]	37.4k	52.9k	—	0.987	grokking
seed 2 (wd0.1_s2)[1]	61k	97.6k	—	0.954	grokking

Run	t50 (test-0.5)	Grok time	Width	Final acc (mean of last 5 evals)	Class
i0_d0	39.5k	85k	48k (pop. max)	0.962	grokking
i0_d1	22k	36k	12.5k	0.996 (0.9981 hardened)	grokking
i0_d2	36k	103.5k	45.5k	0.947	grokking
i0_d3	21.5k	28k (pop. min)	8.5k (pop. min)	0.992	grokking
i0_d4	60.5k	72k	15k	0.992	grokking
i0_d5	63k	71.5k	17k	0.970 (0.9513 hardened)	grokking
i0_d6	98k	110k	20k	0.969	grokking
i0_d7	92k	104k	15k	0.983	grokking
i0_d8	66k	89.5k	38k	0.992	grokking
i0_d9	75.5k	94k	20.5k	0.963	grokking
i1_d0	42.5k	135.5k (pop. max)	14.5k	0.936; post-grok regression val 0.95 $\rightarrow$ 0.93 (A.4)	grokking
i2_d0	56.5k	64.5k	13k	0.989	grokking
i3_d0	41k	censored (budget end, mid-transition)	20.5k[2]	0.919 (0.922 at censoring)	censored (A.3)
i4_d0	1.5k (transient; A.4)	118.5k	9k (final rise 115k–124k)	0.995	grokking
i5_d0	78k	89k	27.5k	0.988	grokking
i6_d0	13.5k	29k	18k	0.999 (peak 1.000 @49k; 0.9834 hardened)	grokking
i7_d0	76k	85.5k	11.5k	0.974	grokking
i8_d0	58k	78k	16k	0.975	grokking
i9_d0	119k	censored (budget end, mid-transition)	47k[2]	0.746 (0.748 at censoring)	censored (A.3)

[1] The three early full-resolution film runs (100k budget) are not part of the 19-run population statistics, and they predate the validation split: their diaries carry no val series, so their grok times of record (54k / 52.9k / 97.6k) use train ≥ 0.99 and *test* ≥ 0.95. An earlier draft's table conflated the film's "seed 2" (wd0.1_s2: t50 61k, grok 97.6k) with the population run i2_d0 (t50 56.5k, grok 64.5k); every "seed 2" quote in the prose refers to wd0.1_s2, and the two runs now have separate rows.

[2] Censored runs' widths are reported but excluded from the population width statistic.

[3] Convention note: Section 2's hardening deltas ("within 1.7 points") compare each hardened program to its run's *last-eval* soft accuracy (0.9849 / 0.996 / 1.000 / 0.967 for the quartet), not to this table's mean-of-last-5 column — the two soft conventions differ by at most 0.3 points (the flagship reads 0.985 as a mean of the last five evaluations against 0.9849 at its last evaluation, a gap of 0.0001), and both are stated here once.

Two cohort labels recur in the main text and are disambiguated here once. The range **28k–135.5k** is the grok-time statistic over the 17 grokked runs. The range **22k–79k** (Appendix D.4) is a *different* statistic on a *different* cohort: first val-0.5 crossings over the nine diaried runs (i0_d0–d5, i1_d0, i2_d0, i5_d0; exactly 22,000–79,000; the test-side equivalent is 21.5k–78k). Both are correct; only their labels distinguish them, and one run exhibits both at once: the film run wd0.1_s2 ("seed 2", outside the 19-run population) has t50 61k and grok time 97.6k — two timing statistics, 36k apart, in a single trajectory.

Sensitivity of the first-crossing convention. The grok time has no holding clause, and A.4 records that many runs later fall back below 0.95 validation — i0_d9 to 0.360, i7_d0 to 0.583, i8_d0 to 0.743, i6_d0 to 0.770. In a substrate built on exponentials and logarithms that raises a fair question: is a first crossing catching a transient excursion rather than entry into a generalizing basin? We therefore recomputed the population under a windowed criterion requiring validation ≥ 0.95 to hold for three and for five consecutive evaluations. **The set of grokking runs does not change: 17 under all three criteria.** What changes is the timing. Seven of the seventeen cross later under the five-eval hold — i0_d9 by 22.5k, i0_d2 by 19.5k, i0_d5 by 13.5k, i7_d0 by 3.0k, and i0_d0, i0_d6 and i1_d0 by 1.0k–1.5k — moving the population median from 85.5k to 88.5k and the range from 28k–135.5k to 28k–137k. The post-grok dips are therefore real and sometimes deep, but they are excursions from a basin the run has already entered rather than the crossing itself being spurious. We keep the first-crossing convention because it is the one every timing in this paper was computed under, and record here that the qualitative population picture is unchanged and the median moves by 3k. `scratch_grok_sustained.py` recomputes this from the released metrics.

A.2 The frozen width convention

For each run, **final** = mean test accuracy over the last five logged points, and **plateau** = median test accuracy over steps 2k–10k. The width runs from the last logged point at or below plateau + $0.1\cdot$(final − plateau) preceding the first crossing of plateau + $0.9\cdot$(final − plateau), to that crossing — a last-below-to-first-above rule that measures the *final* rise and is robust to transient early crossings. Under this convention, widths over the 17 grokked runs span 8.5k–48k steps, median 16k. The convention was frozen before the population was scored and applied unmodified to every run, including the non-monotone case of A.4.

A.3 The four-phase taxonomy

Sorted into the four-phase taxonomy of (Liu et al., 2022):

Class	Count	Detail
Grokking	17	generalization delays 21k–129k (delay = grok time minus the first step with train ≥ 0.99; i0_d3 21k to i1_d0 129k)
Censored at budget end	2	mid-transition at cutoff (0.922, 0.748); (Liu et al., 2022)'s strict four-way labels would file them under memorization, which mislabels a 92.2% run mid-jump — we report them as censored instead
Comprehension	0	—

The 92.2% run (i3_d0) ended censored mid-transition and is classed by trajectory shape, not final accuracy; the second censored run (i9_d0) ended at 74.8%. Neither is counted as a grokking failure or a memorization endpoint.

A.4 Two per-run footnotes, in full

i1_d0 (post-grok regression). i1_d0 is counted among the 17 grokked runs under the first-crossing criterion but later regressed from val 0.95 to 0.93 (final test 0.936). Requiring the *final* evaluation to reach 0.95 as well leaves 16 of 19, i1_d0 being the only grokker to miss it. A literal hold-through-budget criterion would be far stricter and we do not use it: 11 of the 17 grokkers dip below val 0.95 at some point after their grok, several far deeper than i1_d0's 0.932 — i0_d9 to 0.360, i7_d0 to 0.583, i8_d0 to 0.743, i6_d0 to 0.770 — and that criterion would leave 6. We report the first-crossing count and the regression rather than choosing the criterion after seeing it.

i4_d0 (non-monotone). i4_d0 rises to 0.90 test accuracy by 10k steps, erodes to 0.69 by 100k, then re-groks to 0.996. It is the source of the transient 1.5k t50 crossing in A.1, and its 2k–10k plateau estimate is contaminated by the early rise. The convention's last-below rule still isolates its final rise cleanly — width 9k, over steps 115k–124k — and we report it with this footnote rather than excluding it.

A.5 HYPERPARAMETERS

The unit forward pass, written out. Each hidden unit takes two arguments. For argument a the unit holds a chooser over the candidate signals available at its layer (25 at layer 0, 73 at layer 1) and forms the convex combination `a_hat = sum_s softmax(logits_a / T)[s] * signal[s]`, with `T` the anneal temperature and two such terms per argument at gains `w` and bias `b`. The unit returns `exp(clip(x_hat, max=x_max)) - ln(softplus(y_hat) + eps)` with `x_max = 8.0` and `eps = 1e-6`. The clip bounds the exponential branch, and the softplus wrapper keeps the logarithm's argument positive — which matters on the very first backward step, since the raw primitive is undefined for `y <= 0` and a randomly initialized network puts mass there immediately. Hardening replaces each softmax by its argmax; the safeguards are unchanged by it.

Hyperparameter	Value
Architecture (modadd)	width 48, depth 2, embedding dim 24, 2 terms per argument; 28,801 parameters
Architecture (sparse parity)	the same substrate and the same width 48, depth 2, embedding dim 24 — only the input arity differs (40 bits rather than two one-hot blocks), giving a 24×40 embedding and 20,450 parameters. The component definitions and every probe therefore carry across unchanged.
Architecture (MLP control)	ReLU MLP, width 96, depth 2, 37,441 parameters, trained on the same p = 97 modular addition at the same optimizer settings. The two MLP censuses (32 and 28 runs) are drawn from one 32-run population.
Task	modular addition, p = 97 (second battery at p = 113), one-hot pairs, 50% train split; (40,3)-sparse parity, 2,000 samples, 50% split
Training budget	150k steps (population runs; flagship seed-0 run: 100k)
Optimizer	AdamW, full-batch, lr 10^{-3}, weight decay 0.1
Training temperature	T = 0.5 (chooser softmax)
Hardening	30k steps; anneal T $0.5 \to 0.01$; selection-entropy penalty 0.05; straight-through last 25%
Numerical safeguards	soft-clamp x_max = 8.0 (the tree module's default, which the training script does not override); softplus $\varepsilon = 10^{-6}$

A.6 MASTER TABLE OF QUOTED NUMBERS

Every number quoted in the main text, grouped by section, with corrected values where the released film data superseded earlier readings (the parity film's norm trajectory — corrections detailed in Appendix D).

Quoted value	What it measures	Where quoted
17/19 grok; censored at 92.2% and 74.8%	population reliability	abstract, Section 1, Section 2
grok times 28k–135.5k (4.8×), median 85.5k; widths 8.5k–48k, median 16k	population timing spread	Section 2, Section 6, Fig. 4

Quoted value	What it measures	Where quoted
71% of non-DC embedding power (69%/73% per input block) in 6 shared frequencies {15, 24, 25, 30, 35, 48} — fundamentals {15, 24, 35} plus aliased harmonics 30 (2·15), 48 (2·24), and 25 ($3 \cdot 24 \equiv -25 \bmod 97$). Verified at the donor checkpoint used throughout: at 100k the two input blocks' top-6 sets are **identical**, {15, 24, 25, 30, 35, 48}, carrying 68.7% and 73.2% of non-DC block power. D.3's sharing *rung* is a different statistic on a per-checkpoint diary convention and flickers at five-of-six; it does not contradict this. Convention of record: power spectrum over frequencies 1–48, DC bin excluded, both input blocks pooled — the convention Figures 2b and 4a compute. (The research log's earlier 84–86% used a per-block top-6 that *included* each column's DC offset, set {0, 15, 24, 30, 35, 48}; retired.)	wave-bank validation	Section 2, Fig. 4b
total parameter norm $\sim$151 @27k $\rightarrow$ $\sim$110 @60k; embedding norm $\sim$79 @20k $\rightarrow$ $\sim$38 @80k	rise-then-collapse profile, both conventions; flagship run only. The arc is **not monotone after 80k**: recomputed from the checkpoints the flagship's embedding norm reads 79.53 @20k, 37.62 @80k and 43.66 @100k, which is the $\|D\|$ that B.11's geometry uses — the same quantity at three times, not two conventions	Section 2, B.11
embedding norm @20k across the 19-run population: min 48.14 (i4_d0), max 74.01 (i5_d0)	population spread quoted in Section 2; the flagship's $\sim$79 (79.53) is outside it because the film runs are excluded from population statistics (A.1 fn. 1). Recomputed from `runs/grok/emltree_*/ckpt_0020000.pt` as the Frobenius norm of `embed.weight`	Section 2
none 0.240 / core-only 0.486 / embed 0.932 / embed+random 0.940 (3 random draws), 0.947 (non-critical donor) / embed+core 0.950	headline five conditions, 2k fine-tune	abstract, Section 1, Section 3.1, Fig. 1
host's own t50 35.4k; own 0.932-crossing at 50.4k	run-specific counterfactual	Section 1, Section 3.1, Fig. 1
core lead median 221 steps, 8 of 8 hosts of one trajectory (autocorrelated; consistency, not a p-value)	circuit-identity stopwatch, kills #24 then #31 (App. B.3)	Section 1, Section 3.1
frozen-embed 0.763; rescued wiring L0 0.78 / L1 0.73 to self; to donor L1 0.02, L0 0.35	division of labor	Section 3.1
$\approx$2,080 bits (260 bytes) vs 4,656 + 4,656 (+97) + 576 dense parameters	capacity accounting	Section 3.1, Section 6
head only (capacity-matched, 4,656 weights) 0.554@2k, $t_{0.9}$ 3.79k, 3.1$\times$ the embedding's	capacity control	abstract, Section 3.1, B.8

Quoted value	What it measures	Where quoted
embed 0.932@2k → 0.990 plateau; complement 0.745@2k → 1.000@17.5k; wiring 0.126@2k → 0.977@20k; none 0.240@2k → 0.968@20k; $t_{0.9}$ 1.2k / 4.9k / 9.6k / 12.7k	symmetric battery (App. B.8)	abstract, Section 3.1, Fig. 2d
host A: none 0.105 / own-future 0.659 / other-run 0.929 / random-at-donor-norm 0.057 / own-at-donor-norm 0.119; host B: none 0.182 / own-future 0.961 / random-at-donor-norm 0.006 / own-at-donor-norm 0.185 (B.5 lists these rows in a different order)	norm-control battery	Section 3.2, App. B.5
norm-control range 0.006–0.533 (six-implant unified convention); random-direction subset 0.006–0.227, host-normed within 0.014 of baseline, random-normed at or well below it	unified convention (App. B.5)	abstract, Section 3.2
donor norms 35.83 vs 45.19 (26% larger)	own-vs-other confound flag	Section 3.2
donors 0.932 / 0.969 / 0.499	donor diversity, never averaged	Section 3.3
p = 113: 0.924 / 0.374 / 0.227	second-prime triplet	Section 3.3, App. B.5
MLP: 0.440 / 0.862 / 0.990 / 0.005 / 0.365	cross-substrate half	Section 3.4, Fig. B1
transformer (p=113 host @10k, 0.256): embed $t_{0.9} \approx$ 0.2k / rand-normed ≈ 3.9k (destructive to ∼500–700, above baseline after) / none $>$ 4k	third substrate, graded (kill #28)	Section 3.4, App. B.9
parity grok 115.5k; ripe host: embed $t_{0.9} \leq$ 0.25k (kill #26 retired the coarse-grid 0.999@4k reading), rand-normed 0.584 anti-rescue; deep host: nine cells (App. B.10)	second task family	Section 3.5, App. B.10
dose-response 0.913 / 0.910 / 0.930 / 0.941 vs 0.135 / 0.132 / 0.240 / 0.828 (independent harness; 30k reads 0.930 vs Section 3.1's 0.932, inside the ±1-pt epsilon)	rescue vs host maturity	Section 3.3, Fig. B1
horizons: embed 0.932@2k → 0.984@6k; none 0.968@20k; host-normed 0.533 → 0.905; rand-normed 0.011 → 0.346	compression, not head start	B.9, Fig. 2c
hardened 98.36 vs soft 98.49 (last eval); +30k soft continuation 97.58; random-init hardening 8%	fidelity + both confounds	Section 2
quartet 0.9836 / 0.9981 / 0.9834 / 0.9513 (range 95.1–99.8%, costs to 1.7 points); agreement 0.9895 on all 9,409 inputs	translation epsilon	Section 2, Section 6
i0_d6 @30k pre-grok donor at donor norm: **inert**, 0.249 @20k, below the random-direction control's 0.346	open anomaly: a learned embedding failing where the class theory says it should not	B.11
f35 joint ablation 0.9918 → 0.9896 (cost 0.0022); f15 (9 dims) 0.9918 → 0.0049	vestigial vs distributed family	App. C.3
ladder 19/19 commensurate (two-rung); last-change 14/19 withdrawn, kept in D.3; seed-2 rungs 78k/83.5k/88.5k	population ladder	Section 4.1, Fig. 3a
L0 match-to-final touches 1.000@98k (0.995 at 100k — one unit flickers at 99k, restored by hardening); L1 0.057@50k → 0.672@100k	the ratchet	Section 4.1, D.3
frequency competition decided steps 400–1000; suppression 9/9, troughs $-0.07\ldots-0.25$; midpoints (val-0.5) 22k–79k	competition/suppression diaries	App. D
cold-restart flip 0.19 → 0.716 (50k → 60k); L0 margin 0.469 → 1.462	flip mechanism	App. D
rescued-network print hard 0.8816 vs soft 0.8829 (≈88%)	printability at consolidation age 2k	App. B.7

B TRANSPLANT BATTERY IN FULL

B.1 THE CORE SET AND THE HEADLINE FIVE CONDITIONS

"Core circuit units" are the k = 6 layer-0 units whose single-unit ablation in the trained donor costs more than $\delta = 0.05$ test accuracy on the final soft checkpoint — the set {18, 19, 28, 42, 44, 45}; "core-only" and "embed+core" implant exactly this set. The headline battery operates on the flagship run's own history: donor = emltree_wd0.1_cpu at 100k steps, host = the same run at 30k steps (test 0.203), each condition followed by a 2,000-step full-batch fine-tune of everything:

Condition (seed-0 host @30k, 2k fine-tune)	Test accuracy
No implant (fine-tune only)	0.240
Core units only (the k = 6 set)	0.486
Embedding only	0.932
Embedding + random replacement units (3 draws), then non-critical donor units (1)	0.940, 0.940, 0.940; 0.947
Embedding + donor core units	0.950

A *random replacement unit* is a core-position unit whose chooser logits are copied from a freshly constructed, **randomly initialized** network (seeds 500/501/502) rather than from the donor; a fourth cell, reported in the same row, instead implants six arbitrary *non-critical donor* units and is a different condition, which we state rather than pool: the arm asks whether the donor's learned circuit identity matters at all, not whether one donor unit can stand in for another. One provenance note, because the released log is easy to misread: the 0.947 cell appears in `runs/transplant.log` under the label `embed+rand`, where *rand* abbreviates *randomly chosen* and not *randomly initialized*. It is `rung3_transplant.py`'s `RAND = [1, 3, 8, 20, 31, 40]`, six arbitrary non-critical units copied from the *donor*, as that script's transplant routine and its docstring both state. The three randomly initialized draws are the later `runs/fix_battery.log` at 0.940 each. The label, not the condition, is what changed. The three random-unit draws are reported individually, never as mean ± sd — n = 4 with a degenerate distribution makes a standard deviation misleading; the claim is that all four sit inside the ±1-point epsilon band of B.2, beside embed+core's 0.950.

B.2 MEASUREMENT EPSILON: THE NEIGHBORING-CHECKPOINT BOOTSTRAP

Training and fine-tuning are full-batch, so fine-tunes are deterministic given a fixed numerical configuration; the honest scale for a small between-condition gap is measurement variability under ancillary choices, and we measured it directly by shifting the host checkpoint ±1k steps. Embed-only at hosts 29k / 30k / 31k: 0.941 / 0.932 / 0.938. A core-augmented variant — donor core logits copied at *both* layers, a superset of the Section 3.1 layer-0-only embed+core condition, which is why it reads 0.957 rather than 0.950 at the 30k host — gives 0.962 / 0.957 / 0.966 — margins of 0.021, 0.025 and 0.027, the third computed from the unrounded $0.9656 - 0.9382 = 0.0274$ rather than from the three-decimal figures displayed here, which would round to 0.028. Each condition spans 0.9 points under a ±1k host shift; that spread is the measurement resolution quoted in Section 3.1 (embed-only at the 30k host reproduces 0.932 exactly, validating the harness). That spread measures sensitivity to *which* checkpoint is the host, not to the arithmetic, and for a host that has not yet committed the two are not the same size. Re-running B.11's boundary-phase cell − host i0_d1 at 0.913, donor i2_d0, every quantity identical − under a different BLAS thread count moves the embed-minus-none gap from −0.160 to −0.214, because the reduction order in the matrix products changes and a host in mid-transition amplifies the difference over 2,000 steps. Hosts that have committed do not amplify it: the same change moves the deep post-grok cell by 0.0007 and the near-ceiling cell by 0.0015. We therefore read the 0.9-point figure as the resolution for settled hosts and treat boundary-phase cells as carrying a wider envelope, and the thread count is recorded per script for the batteries registered from #39 onward; the five headline scripts predate that convention and pin no thread count, so their cells carry the wider envelope. For the cross-run batteries of B.5, the error bar is the cross-host spread, which is why we report two hosts per condition rather than fine-tune seeds.

B.3 CIRCUIT IDENTITY AT STOPWATCH RESOLUTION (KILL #22)

The 2k-endpoint comparison of B.1 (0.950 vs. 0.940–0.947) reads circuit identity as at most one point — but that endpoint sits near ceiling, and the registered follow-up (evaluate every 100 steps; report t50 and t90 of the fine-tune trajectory — t90 here is the same statistic as the main text's $t_{0.9}$, at 100-step resolution) resolves what the endpoint cannot:

Condition	t50 (steps)	t90 (steps)	Final @2k
Embedding + donor core	400	900	0.957
Embedding + random core, three draws (seeds 500/501/502)	600 (all three)	1,100 (all three)	0.938–0.942
Embedding only	600	1,200	0.932

Three independent random draws land on *identical* timings at 100-step resolution — the draw-spread noise floor is approximately zero — and against that floor the registered no-separation prediction was falsified (kill #22). But draw spread is not the right error bar, and the follow-up host-shift bootstrap (registered: withdraw if the spread reaches 200 steps) settles it:

Host	embed_only t50/t90	embed_core t50/t90	embed_randcore t50/t90
29k	600 / 1,300	400 / 900	600 / 1,200
30k	600 / 1,200	400 / 900	600 / 1,100
31k	400 / 900	300 / 600	400 / 800

Within every host the donor core leads random replacements — 200–300 steps at t90, same sign 3/3 — but the same timings shift 300–400 steps under a ±1k host shift, exceeding the paired advantage, and per the registered criterion the timing claim was **withdrawn (kill #24)**. That withdrawal compared a *paired* effect against an *unpaired* spread, which is the wrong test; the redo below runs the paired design at adequate n and reinstates the effect. **Kill #24 is itself withdrawn (kill #31).**

The eight-host paired redo (kill #31). Registered before launch with the statistic and both thresholds fixed: d_h = mean t90(random core over three draws) − t90(donor core), with P1 holding if the median d_h $\leq$ 100 steps and at most 6 of 8 hosts favour the core, and P1 falsified at $\geq$ 7 of 8 with a median $\geq$ 150. One condition note, because the distinction matters: the core arm here copies the k = 6 units' chooser logits at *both* layers, so it is B.2's superset condition (0.957 at the 30k host, reproduced exactly by this battery) rather than Section 3.1's layer-0-only embed+core (0.950). The comparison is internally matched — donor and randomly initialized replacements occupy the same positions at the same layers — but the arm is not the headline five-condition battery's, and the embed-only arm reproducing Section 3.1's 0.932 to four places is the check that the harness is otherwise the same. Eight plateau hosts of the flagship run, five arms each, evaluated every 25 steps.

Host	frozen acc	donor core t90	random-core mean	lead	lead as share
24k	0.150	675	1,625	950	58%
26k	0.160	800	1,192	392	33%
28k	0.167	825	1,208	383	32%
29k	0.182	900	1,158	258	22%
30k	0.203	900	1,083	183	17%
31k	0.248	575	750	175	23%
32k	0.289	625	767	142	18%
34k	0.406	625	783	158	20%

The donor core leads on **8 of 8** hosts with a median lead of **221 steps**. We state the inferential caveat plainly, because it is ours: these eight hosts are checkpoints 24k–34k of *one* deterministic trajectory,

not eight independent runs. They are autocorrelated, so the nominal sign-test and exact-Wilcoxon p of 0.0078 (the two coincide here by construction, since all eight differences share a sign and W is maximal) overstates the evidence, and the effective n is smaller than eight. What the sweep does establish is consistency across host maturity — a lead present at every one of eight maturities spanning 0.150 to 0.406 frozen accuracy, at 17–58% of the arm's own crossing time — and that, not the p-value, is the claim we make. A cross-run replication of the headline *timing* was the natural next test, and it has now been run; B.3's caveat is what motivated it. Its predictions, falsification thresholds and censoring rule were committed in the docstring of `scratch_crossrun_t90.py` before any result existed. It applies the headline design unchanged to three **independently initialised** lineages, each serving as its own host and donor: host at 30k, donor at 100k, an embed-only arm against a fresh-optimizer no-implant baseline, $t_{0.9}$ read on a 250-step grid to 20k, threads pinned. The three lineages were fixed in advance on two stated criteria — grok comfortably after 30k, so the host is genuinely stuck, and before 100k, so the donor is grokked — which excluded the i0_d* family (shared initialisation), i6_d0 (already grokked at 30k), and i3_d0 and i9_d0 (never grok). The embedding arm crosses at **1,000 / 2,000 / 2,500** steps on i2_d0 / i5_d0 / i7_d0, reaching 0.9862 / 0.9826 / 0.9983 by 20k, while **none of the three baselines reaches 0.9 at all** inside 20k, ending at 0.4385 / 0.6616 / 0.0948. All three registered predictions are met on 3 of 3 lineages, against thresholds requiring 2 of 3: embed $\leq$ 3,000 (P1); baseline $\geq$ 8,000, which a censored baseline satisfies by the registered rule (P2); and compression $\geq 3\times$, which the same rule reads as $\geq 20\times$ / $\geq 10\times$ / $\geq 8\times$ (P3). Those three factors are lower bounds set by the 20k budget rather than measurements, since no baseline crossed inside it; the flagship's $10.5\times$ (12,713 against 1,215, both interpolated) remains the only measured compression, and extending these baselines to their crossings is the obvious next run. The independent lineages are therefore *harder* than the flagship, whose own baseline does cross, at 12,713 — so the headline compression is not an artifact of one trajectory. What this battery does **not** replicate is the core-identity lead of this subsection: it ran the embed and no-implant arms only, so the 8-of-8 donor-core result above still rests on checkpoints of one trajectory, and we leave that caveat standing. `runs/crossrun_t90.jsonl`; the i7_d0 baseline cell was finished afterwards by `scratch_crossrun_t90_i7base.py`, which repeats the registered design for that one cell and appends to the same file. The result survives each robustness

check we ran: dropping the 34k host, whose 0.406 frozen accuracy puts it inside its own transition, leaves 7 of 7 (p = 0.0156, median 258); scoring the donor core against the *fastest* of the three random draws rather than their mean leaves 8 of 8 (median 212); and the donor core also beats leaving the host's own units in place, 8 of 8, median 288. P1 is falsified on the criterion fixed before the run.

Two further readings the redo corrects, both of which we had asserted from the three-host data. The effect is roughly a constant *fraction* of the residual rebuild rather than a constant number of steps — the relative lead sits at 17–33% on seven of the eight hosts, the youngest (24k, 58%) being the outlier — so the apparent decline of the absolute lead with host maturity is mostly the ceiling effect of every arm finishing sooner on a riper host, not a dose-response. And random replacement units are *not* generally harmful: they beat the host's own units on six of eight hosts (median -79 steps), harming only on the two youngest, so the arm is a near-neutral control rather than a destructive one.

The claim of record. Circuit identity is worth a median 221 fine-tuning steps: about 1.9% of the $\sim$11.5k the embedding buys, and about a fifth of what remains to be rebuilt once the embedding is supplied. Small, real, consistently signed, and reported rather than rounded to zero. B.1's endpoint null stands unchanged and is now explained rather than merely reported — at the 2k read every arm is near ceiling, so circuit identity changes *when* the host arrives, not *where*.

B.4 CAPACITY ACCOUNTING

The two sides of the dissociation do not have comparable capacity, and the accounting is part of the record. The wiring's entire discrete hypothesis space comprises 384 selections — 48 units per layer $\times$ 2 layers, each unit choosing 2 signals for each of its two arguments — with layer 0 selecting among 25 options (24 embedding coordinates plus the constant 1) and layer 1 among 73 (48 unit outputs, 24 re-exposed coordinates, the constant): $\log_2(25^{192} \cdot 73^{192}) \approx 2{,}080$ bits, so every possible hardened wiring fits in about 260 bytes. The dense maps live in continuous spaces of far greater capacity: 4,656 embedding parameters, 4,656 head weights (plus 97 biases), and 576 per-unit affine parameters. On raw accounting, "circuits are cheap" could be an architectural artifact. Two measurements convert the limitation into part of the measurement: the frozen-embed condition (B.7) shows a working circuit rebuilt from scratch on fixed features within 2k steps — a fact about the optimization landscape given the representation, not about the count of wirings — and the converse is measured too: the structureless norm implant sits at 0.346 after 20,000 steps, below the no-implant baseline at every horizon (B.9). The accounting says the small component *could* be cheap; the experiments show that it is, and that the large one is genuinely slow. A third measurement closes the objection at matched capacity rather than by argument: the readout head carries exactly the embedding's 4,656 weights and the same spectral character (C.4), and implanted alone it is 3.1$\times$ slower to rescue ($t_{0.9}$ 3.79k against 1.2k; B.8). Two dense maps of identical size, opposite ends of the computation, different rebuild times — capacity does not order the components; position in the computation does.

B.5 HOSTS A AND B: THE NORM-CONTROL BATTERY AND THE RANGE CONVENTION

Host A is a different run (i7_d0) frozen at 40k steps, at 7.4% test accuracy; host B is a second cross-run host (i0_d5) at 30k steps, at 12.4%. All conditions receive the same 2k fine-tune:

Condition	Host A (i7_d0 @40k)	Host B (i0_d5 @30k)	Modulus-113 host (@10k)
No implant (fine-tune only)	0.105	0.182	0.374
Grokked embedding, other run	0.929	—	0.924
Own future embedding (embed_OWN)	0.659	0.961	—
Random direction at donor norm	0.057	0.006	0.227
Own embedding at donor norm	0.119	0.185	—

The third column is the modulus-113 replication of Section 3.3, whose host reads 0.296 when frozen.

One labeling note, corrected against the Gate-1 log: host B's 0.961 is the **embed_OWN** condition — the host run's *own* future embedding — not an other-run donor; the table above and Figure 2a carry the same assignment (an earlier draft's table had misplaced it, and the log is the authority).

Range convention. Every norm-control range quoted in this paper is the min–max over *all six* EML norm-matched implants — random-direction and own-embedding rescaled to donor norm on hosts A and B at modulus 97, the random-normed control at modulus 113, and the flagship host's own embedding at donor norm (B.9) — i.e., **0.006–0.533** at the 2k read. The sixth cell is the outlier and it is the informative one: rescaling the *flagship* host's own embedding reaches 0.533 at 2k and plateaus at 0.905, where the same operation on the younger hosts A and B lands at 0.119 and 0.185. The difference is host maturity — the 30k flagship host has real partial structure to amplify, hosts A and B do not — and it is the reason the claim of record is about *random-direction* implants, which fail or harm on every host, rather than about norm-matching in general. The own-embedding-at-donor-norm controls land within 0.014 of their baselines (the one control above baseline is host A's, 0.119 against 0.105); the random-direction controls land at or well *below* baseline — by as much as 0.176 (host B, 0.006 against 0.182) — the norm-without-structure harm of B.9 already visible at 2k. The MLP's norm controls (0.005, 0.365) are quoted separately because its baseline (0.440) differs, and the 20k-horizon readings of B.9 separately again, because that battery runs on a different host and horizon.

Donor norms. The own-versus-other ordering on host A carries a flagged confound: final embedding norms are 35.83 (i7) against 45.19 (the other-run donor) — the other run's donor is 26% larger in norm, so the 0.929-over-0.659 gap does not isolate frequency identity, and we build nothing on it.

B.6 DONOR DIVERSITY

All three grokked donors tried against the seed-0 host beat the 0.240 baseline; two of three rescue fully. Per-donor rows, with metadata, never averaged:

Donor	Rescue (seed-0 host @30k, 2k)	Donor grok time	Donor final accuracy	Donor hardened
seed-0 self (own future @100k)	0.932	54k	0.9849	0.9836
i6_d0	0.969	29k	1.000	0.9834
i0_d1	0.499	36k	0.996	0.9981
No implant	0.240	—	—	—

The 0.499 sits in plain view: i0_d1 rescues to twice baseline but not fully. At n = 3 this was an open question; B.11's 8-pair battery resolves part of it — i0_d1 rescues a different host normally (0.634 over 0.111), so the 0.499 was pair-specific — and honestly complicates the rest (a floor effect, not a predictor).

B.7 THE FROZEN-EMBEDDING CONDITION, THE RESCUED NETWORK'S WIRING, AND ITS PRINT

Freezing the implanted embedding still rescues, to **0.763** (against 0.932 trainable and 0.240 baseline) — the circuit rebuilds from scratch on *fixed* features, and refinement of the implant supplies the rest. The circuit that does the rebuilding is not the donor's. The network read here is the embed-plus-random-core arm (0.940), not the embed-only rescue, so six core layer-0 units carry random logits in it; the read is of the *host's* retained wiring either way. It retains its own pre-rescue wiring — layer-0 match to its pre-rescue self **0.78**, layer-1 **0.73**; to the *donor*, layer 1 retains only **0.02** but layer 0 retains **0.35** — "any circuit will do" includes the host's own, retuned. And the rescued network's print: pressed with the unmodified hardening recipe, it hardens faithfully — hard **0.8816** against soft **0.8829** — but at $\approx$88%, well below the $\sim$98% prints of the long-consolidated networks of Section 2: printability tracks consolidation age, here set by intervention.

The mirror condition, and the asymmetry it exposes (registered). B.7 freezes the donor's *embedding* and lets the circuit rebuild. The complementary arm freezes the donor's *wiring and head* and lets only the host's own 30k embedding move. We registered that it would be slower than B.7's rebuild, falsified if it crossed 0.9 at or before 2k. It is very much slower: **0.037 at 2k**, 0.5 at 6,750, and it **never reaches 0.9 within 20,000 steps**. Given a mature representation the circuit rebuilds to 0.763 in 2k; given a mature circuit the representation does not rebuild at all on this horizon. The two arms freeze a donor component and train the rest, so the comparison is matched, and it is the sharpest statement of the paper's asymmetry we have: what is expensive here is the representation, not the routing that reads it.

B.8 THE SYMMETRIC BATTERY ON MODULAR ADDITION

The strongest form of the capacity objection is the missing converse, and it was run as a registered experiment (prediction logged before launch: "genuine uncertainty, prior leans rescue"): implant everything *except* the embedding, over a 20k-step horizon, with an early evaluation grid at 100/250/500/1k/2k steps. All four arms share the fresh-optimizer convention of B.9; determinism checks passed (embed 0.932 at 2k, none 0.240 at 2k — the paper's exact numbers, and the none arm is byte-identical to the B.9 horizon run's none arm).

Condition (seed-0 host @30k)	@2k	Endpoint	$t_{0.9}$
Embedding only	0.932	0.990 plateau	1.2k
Complement (donor layers + head, host embedding)	0.745	1.000 @17.5k	4.9k
Head only (capacity-matched dense control)	0.554	0.992 @20k	3.79k
Wiring alone (chooser logits only)	0.126	0.977 @20k	9.6k; below baseline before $\sim$2.5k
No implant (fresh optimizer)	0.240	0.968 @20k	12.7k

$t_{0.9}$ is the paper's one timing statistic: the first crossing of test accuracy 0.9 during fine-tuning, interpolated on the evaluation grid here and in B.3's eight-host sweep. B.3's cross-run replication is the exception and reports grid-first crossings (1,000 / 2,000 / 2,500; interpolating gives 821 / 1,819 / 2,370), which is the conservative direction and is why we left it. ***Four arms are measured on a 250-step grid to 20k; the head arm is on that grid to 6k and on a 2,000-step grid beyond it*** *(interpolated crossings, in the table's row order: embedding 1,215, complement 4,901, head 3,788, wiring 9,612, none 12,713). The head arm alone had been re-gridded, and the head-versus-complement gap was smaller than one interval of the coarser arm, so the finer grid was extended to it. The ordering is unchanged and the gap is 1.11k, more than four times the resolution; three crossings moved down by 0.04–0.21k and the embedding's 1.2k is confirmed exactly. One transient is left in the record rather than smoothed: the embedding arm reads 0.614 at step 14,750 between neighbours of 0.990 and 0.992, a single-eval excursion in a deterministic run, reminiscent of (Thilak et al., 2022)'s slingshot and unexplained here. It is not the only one:* `crossrun_t90.jsonl` *carries the same one-evaluation collapse on i2_d0 (0.496) and i7_d0 (0.341), both at step 4,750. Three of the four embedding arms read at this resolution show one and two share a step, which points at the evaluation or the clip/softplus safeguards rather than at training. None falls near a $t_{0.9}$ crossing, so no timing in this paper depends on them, but the pattern is larger than the single case reported above and we flag it as unresolved.*

Host-shift replication (registered). A $t_{0.9}$ is one deterministic measurement on one host, and B.3 established that the honest error source is a ±1k host shift. That bootstrap had been applied to the circuit-identity claim and not to this battery. All five arms, re-run at the neighbouring hosts on the same 250-step grid:

Arm	host 29k	host 30k	host 31k	range
Embedding	1,169	1,215	861	861–1,215
Head (capacity-matched)	3,884	3,788	6,338	3,788–6,338
Complement	6,572	4,901	11,319	4,901–11,319
Wiring alone	8,260	9,612	6,210	6,210–9,612
No implant	$>$ 14,000	12,713	$>$ 14,000	$\geq$ 12,713

2,000-step reads on the shifted hosts. The table above is a timing statistic; the accuracy reads behind Section 3.1's harmful-phase claim are below, from `runs/hostshift.log`. The 30k column is the five-condition table at the head of this subsection.

Arm	host 29k	host 31k
Head (capacity-matched)	0.516	0.695
Complement	0.676	0.718
Wiring alone	0.130	0.438
No implant (baseline)	0.201	0.344

The wiring arm is below its baseline at 29k (0.130 against 0.201) and above it at 31k (0.438 against 0.344) — the contrast Section 3.1 quotes. The embedding arm's 2k reads were not recorded in this battery, only its crossings.

The 14k evaluation budget censors the baseline at two of three hosts. It was set from the 30k baseline's 12,713 with too little margin, which is a fault of the design rather than a property of the runs.

The registered prediction was that the ordering would hold at all three hosts, with falsification if the head and complement inverted **or** if their ranges overlapped. The ranges overlap — head 3,788–6,338 against complement 4,901–11,319 — so **the registered criterion fails**, and we report it as stated rather than reinterpreting it after the fact. We note separately that the criterion was badly chosen: it tests a paired effect against an unpaired spread, which is exactly the error kill #24 made and kill #31 withdrew. Paired within each host, the head crosses before the complement 3 times out of 3. The conclusion we draw is the conservative one: the embedding's primacy and the head-before-complement relation hold at every host, the position of the wiring arm does not, and the five-arm ranking is withdrawn.

Read against the rising baseline — the fresh-Adam host groks alone at $t_{0.9} \approx 13$k, the optimizer-reset effect — the acceleration ordering is the finding: embedding $\sim$11.5k $\gg$ complement $\sim$7.8k $>$ wiring $\sim$3.1k (and harmful early) $\gtrsim$ none. The registered uncertainty resolved toward rescue: the complement does rescue, but the ranking preserves embedding-primacy, now measured from *both* directions — given the embedding, the circuit rebuilds in $\sim$2k; given the entire rest of the network, the embedding rebuilds in $\sim$4.9k. Two consistency checks anchor the battery: the complement at 2k (0.745) lands on the frozen-embedding rebuild (0.763) — two designs agreeing on how far a host gets while the embedding problem is unsolved — and every implant arm dips below baseline at step 100 (any transplant first breaks the memorized component match before rebuilding — at step 0 the embedding implant reads 0.019, *below* the untouched host's 0.203: the implant is not a zero-shot module but scaffolding whose value is realized through adaptation, which is why the paper's statistic is rebuild time under a matched fine-tune, never zero-shot transfer). Figure 2d plots the five trajectories.

B.9 Modulus 113, matched horizons, and the MLP

Modulus 113. A host at 10k steps (0.296) reaches **0.924** with the embedding implant, **0.374** without, and **0.227** with the random-normed control — the norm control lands below baseline at the second modulus too.

Horizons: compression, not a head start. Every number above is read at 2,000 steps, and a fair deflation says the implant merely shifts the schedule forward. It does not. Run the same four conditions for 20,000 steps (Figure 2c, table in B.9) and the untouched baseline does catch up, crossing 0.9 near 12.7k — but the implanted host is above every other condition at every horizon past $\sim$250 steps, and the run's own unaided history did not reach 0.932 until step 50.4k. The gain is compression, not a head start.

The norm controls invert that deflation rather than refuting it. Rescaling the host's own embedding to donor norm buys an early boost and then plateaus *below* where the untouched baseline ends, and a random direction at the same norm is destructive at every horizon. Norm without structure does not merely fail to help; it harms.

Matched horizons (Figure 2c). On the seed-0 host at 30k, four conditions fine-tuned for 20,000 steps, accuracy read every 2k; the host's own unaided trajectory crossed 0.932 at step 50.4k:

Condition	@2k	@20k	Trajectory
No implant (fine-tune only)	0.240	0.968	$t_{0.9} \approx 13$k; crosses 0.932 by $\sim$16k (0.919 at 14k, 0.934 at 16k)
Grokked embedding	0.932	dominant	0.984 by 6k, 0.990 plateau; above every condition at every horizon
Own embedding at donor norm	0.533	0.905	early boost, then plateau below the natural run
Random direction at donor norm	0.011	0.346	below no-implant at every horizon; 62 points behind baseline at 20k

The full 2k-resolution curves are Figure 2c's data, reproduced from a fresh rerun of all four arms (third determinism confirmation: the rerun matches the table exactly).

Figure B1 — MLP transfer and dose-response renders here; its two panels were demoted from Figure 2, their numbers living in the table below and Section 3.3.

The MLP full table. On a clean 5k-step memorizing MLP host (21.1% at the checkpoint):

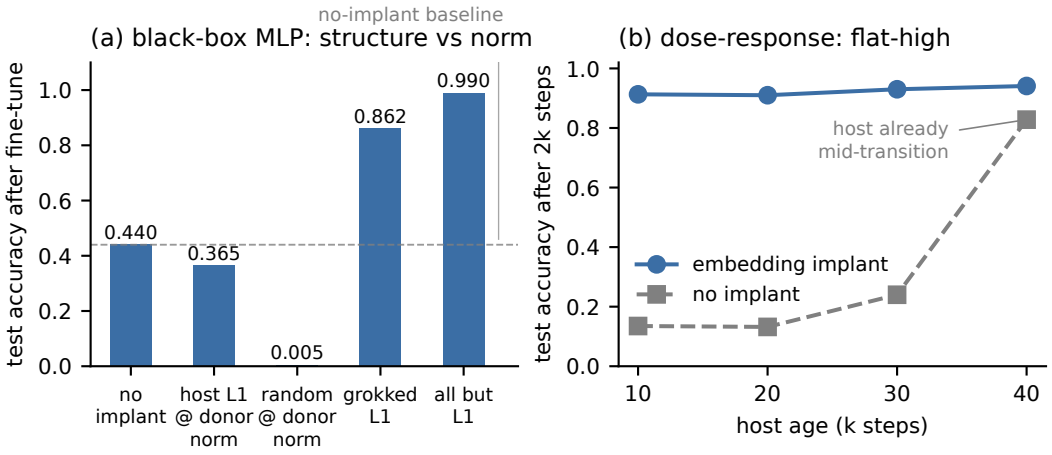


Figure B1: (Appendix B.9) — The demoted panels: MLP transfer and dose-response. (a) MLP: the first layer rescues (0.862) — so does its complement (0.990); the cross-substrate claim is structure-vs-norm only (Section 3.4). (b) Dose-response: flat-high implant (0.910–0.941) against a rising control (0.135 → 0.828; the 40k control point is mid-transition; independent harness — its 30k reading 0.930 sits within the ±1-point epsilon of Section 3.1's 0.932).

MLP condition (2k fine-tune)	Test accuracy
No implant (fine-tune only)	0.440
Grokked MLP's first layer	0.862
Everything except the first layer (all-but-L1)	0.990
First layer, random-normed	0.005
Host's own first layer at donor norm	0.365

The all-but-L1 row is the one that breaks the dissociation in the black box — the complement rescues as fast as the layer itself — and its reading belongs to Section 3.4: an MLP has no architectural seam between representation and circuit, its later layers being themselves representation-bearing, so the experiment cannot separate component rebuild times; the cross-substrate claim is scoped to structure-versus-norm. (In the glass box, where the seam exists, the complement also rescues — but 3–5× slower than the embedding: B.8.)

The transformer battery (registered). A one-layer decoder-only transformer (d_model 128, 4 heads, d_mlp 512; modulus 113, 30% split, AdamW lr 10^{-3}, weight decay 1.0 — the Nanda-style recipe) memorizes by step 500, sits far below its final accuracy through 10k without being at chance there, and groks at ∼16.75k (final 1.000). The battery runs on its 10k checkpoint (test 0.256, which is low but is twenty-nine times chance at p = 113), implanting only the token-embedding matrix, fresh optimizer, evaluated every 100 steps:

Transformer arm (host @10k)	$t_{0.9}$	Trajectory
Grokked embedding (from 40k)	≈ 0.2k	0.854 @100, 0.966 @200, 1.000 by 500
Random direction at donor norm	≈ 3.9k	destructive early (0.043 @100, below baseline to ∼500–700), then accelerating: 0.383 @2k, 0.987 @4k
No implant	> 4k	0.259 → 0.298 @4k

Registered outcome, both halves reported: the structure prediction was confirmed emphatically (near-instant rescue); the norm-control prediction ("at or below baseline") was **falsified at the 2k read (kill #28)** — 0.383 against 0.271 — because in this architecture the random implant, after a destructive opening, eventually un-sticks the host. The structure-over-norm ordering survives at ∼20× (0.2k against 3.9k), and the radial contribution is consistent with (Liu et al., 2023) under this recipe's 10× larger weight decay; the flat claim "norm controls are dead" is therefore scoped to the EML and MLP substrates, and the transformer row is quoted as graded throughout.

B.10 THE SPARSE-PARITY BATTERY IN FULL

(40,3)-sparse parity groks in this substrate at 115.5k steps (snap to 0.998 after 100k+ steps at chance over a perfectly memorized training set). The battery covers two host phases.

The battery's spine. Near the transition structure is necessary; deep in the plateau any rich implant suffices and structure adds speed.

Parity condition	Ripe host @100k	Deep host @60k (20k steps)
No implant	groks naturally, $t_{0.9} \approx$ 7k (fine grid; 8–12k in the earlier harness)	0.513 (flat)
Donor embedding (scale-matched)	$t_{0.9} \leq$ 0.25k (0.991 at the first eval; coarse-grid reading 0.999 @4k)	1.000 @4k
Donor embedding (native norm 1.3)	—	0.479 (starves under wd; see scale-matching law)
Random embedding (norm-matched)	**0.584 — anti-rescue**	0.995–1.000 (any scale)
Donor wiring / complement	$t_{0.9} \approx$ **1.7k / ≈ 3k** (0.984 / 0.994 @20k; B.10)	0.995 / 0.993
Donor wiring, second lineage — taken at grok	$t_{0.9} \approx$ **1.1k** (vs. baseline 5.6k)	—
Donor wiring, second lineage — taken at 150k	9.5k — *below baseline* (cleanup rewired it)	—

Ripe host @100k (15k steps short of its transition):

Condition	Outcome
No implant	$t_{0.9} \approx$ 6.9k on the fine grid (the earlier harness's coarse read was 8–12k; the fine grid is of record)
Donor embedding (scale-matched)	$t_{0.9} \leq$ 0.25k (the coarse grid's 0.999 @4k was a bound, not a timing — kill #26)
Host-normed embedding	0.995
Random embedding, norm-matched	**0.584 @20k — anti-rescue**: destroys natural progress

Deep host @60k, 20k-step horizon — the deep-host grid, in the measured order (with the ripe-host rows above, the battery's eleven distinct conditions):

Condition	Outcome
Donor embedding at host norm	1.000 @4k — fastest of all cells
Random embedding at host norm	0.859 @4k → 1.000
Random embedding at donor-native norm 1.3	0.995
Host's own embedding at norm 1.3	1.000 @16k
Donor wiring	0.995 (snap at 16–20k)
Complement (all-but-embedding)	0.993 by 8k
Zero embedding	0.712 crawl (∼0.62 @16k)
Donor embedding at native norm 1.3	0.479 — starves
No implant	0.513 (flat)

The ripe-host converse battery (registered; the once-missing cells). Run against predictions registered before launch (the ripe host @100k, donor @150k, fresh AdamW, eval every 250 steps to 4k):

Ripe-host arm	$t_{0.9}$	Trajectory
No implant	$\approx$ 6.9k	chance to $\sim$5k, natural grok (0.844 @6.5k $\rightarrow$ 0.928 @7k); 0.994 @20k
Donor wiring (chooser logits only)	$\approx$ 1.7k	snap between 1.25k and 1.5k (0.522 $\rightarrow$ 0.831); 0.984 @20k, peak 1.000 @17k
Complement (all-but-embedding)	$\approx$ 3k	0.931 @3k; 1.000 by 11–12k, 0.994 @20k
Donor embedding (scale-matched, addendum arm)	$\leq$ 0.25k	0.991 at the first evaluation (step 250); 0.996 @8k

Registered outcomes, all reported: the wiring-rescue prediction ($t_{0.9} \leq$ 4k) was **confirmed** — the wiring arm supplies most of the wait with no harmful phase, where modadd's wiring arm was weakest and initially harmful. Two registered predictions were **falsified**: the complement was predicted at least as fast as wiring-only and is slower (3k against 1.7k; reported as observed, no mechanism claimed), and the embedding was predicted at 2–4k, slower than wiring — it is instant ($\leq$0.25k; the earlier "0.999 @4k" was a coarse evaluation grid's bound, not a timing). The instant embedding on a wiring-slow task is the sufficient-scaffolding/natural-order distinction in one cell: parity's training never builds embedding structure before the snap (Figure 3b), yet a supplied one is exploited immediately — what intervention can use and what training builds are different questions, and the battery plus films answer both.

Third and fourth lineages (registered before launch). The battery was repeated unchanged on the two plateau lineages found later, hosts matched to the same $\sim$15k-short convention and donors taken at grok:

Arm	i4, host @103k (grok 118.5k)	i7, host @48k (grok 63k)
No implant	$t_{0.9}$ 8.85k	**never reaches 0.9** (0.640 @20k)
Donor wiring, at grok	$t_{0.9}$ **1.35k**	$t_{0.9}$ **1.20k** (0.978 @2k, 1.000 @10k)
Donor embedding, scale-matched	$t_{0.9}$ 0.35k	$t_{0.9}$ 1.22k
Random embedding at host norm	$t_{0.9}$ 17.5k, 0.993 @20k	never reaches 0.9, 0.496 @20k

Registered outcomes, all reported. **P1 confirmed on both**: donor wiring rescues at $\leq$ 4k, and on i7 the result is categorical rather than a speed-up — the host does not recover unaided within the horizon at all. Across the four lineages the wiring arm's crossing is tight (1.7k, 1.1k, 1.35k, 1.20k) where the baselines are not (6.9k, 5.6k, 8.85k, $>$ 20k). **P2 falsified on i7**: the scale-matched embedding was registered at $\leq$ 0.5k and reads 1.22k. **P3 falsified on i4**: the prediction was that the random implant sits at or below the no-implant arm *at 20k*, and it reads 0.993 against 0.984 — marginally above. It is a genuine anti-rescue in *timing* (17.5k against 8.85k, costing the host some 8,700 steps), but that is not what we registered, and we score the criterion we wrote rather than the one the data would have passed (kill #38).

One cell is worth stating on its own, unregistered and n = 1. On i7 the wiring arm (1.20k) and the embedding arm (1.22k) are **tied**. On every other host in this paper, on either task, the embedding is fastest by a wide margin; here a finished circuit is worth as much as a finished representation. This is also the only host whose donor norm already equalled its own, so the scale-matching step was a no-op.

The second-lineage replication, and the at-grok donor law (registered). The ripe-host battery was repeated on an independent parity trajectory (parity403_i1, grok 95k) at a matched 15k-pre-transition host, with the embedding arm scale-matched per the law below:

Arm (second lineage, host @80k)	$t_{0.9}$
Donor embedding, scale-matched	0.5k
Donor wiring, taken at grok (95k)	**1.1k**
No implant	5.6k
Donor wiring, taken at the final checkpoint (150k)	9.5k — *slower than no implant*

The registered prediction (wiring rescues at $t_{0.9} \leq$ 4k) was **falsified** on the first attempt, which used the 150k donor: that arm finished 1.7$\times$ slower than the untouched baseline. The registered candidate mechanism — drawn from this lineage's own film, which records its layer-0 wiring migrating away

from its at-grok program during cleanup — was then tested and **confirmed**: donor wiring match to its own at-grok program is 0.021 at 150k for this run against 0.495 for the first run, and taking the donor at grok restores the rescue at 1.1k. The first lineage's naive final-checkpoint donor worked only because its cleanup preserved half the solution circuit. Both stages stand in the record (B.12); the operative condition is that a donor *circuit* must be taken at grok, exactly as a donor *embedding* must be taken at the host's operating scale.

Extension and cross-host replications. The 60k-step extension shows donor@1.3 on the 60k host is genuinely inert, not censored (still $\sim$0.50 while zero-embed crawled to 0.712 — a minor open puzzle, noted). Cross-host, donor@1.3 *rescues* at hosts 50k (1.000 @20k; 0.977 @5k) and 70k (0.997) — the 60k failure is checkpoint-specific, disclosed as an open observation; the cross-host ladder is mandatory for single cells, and random@1.3 rescues at both flanking hosts too.

The scale condition, and why we no longer call it a law. Parity's consolidation collapses the embedding norm roughly 18-fold from its mid-plateau peak — a weight-decay erosion through the plateau's second half ($\sim$23 at 40k $\rightarrow$ 14.2 at grok) followed by an $\sim 11\times$ post-grok collapse to 1.28 (Appendix D.5) — so a naive final-checkpoint implant arrives at native norm 1.3 into $\sim$20-norm machinery and is eaten by weight decay 0.1 during fine-tuning before it can seed; at functional (host-matched) scale the same donor structure rescues fastest of all cells. Matched-scale implants are reliable and fastest; native-scale implants are fragile — robustness guidance modadd never exposed, its embeddings not collapsing $\sim$18-fold over consolidation. **We state this as an empirical compatibility condition and not as a law, because our own cells refuse the mechanism.** Decoupled decay at lr 10^{-3} and weight decay 0.1 multiplies the norm by a factor 0.9999 per step — 0.819 over 2,000 steps, an 18% shrink, not dissolution. A *random* embedding at the same native norm rescues the very host the structured donor fails on, and donor@1.3 rescues the flanking 50k and 70k hosts, both reported above. If weight decay were eating native-scale implants before they could seed, neither would happen. What we have is one checkpoint at which a native-scale structured donor fails, under one set of optimizer hyperparameters; the matched-scale prescription is sound practice and the explanation for the failure is open.

Phase logic, restated. Near the transition the transplant discriminates (structure rescues, structureless norm anti-rescues — the modadd-style dissociation, replicated); deep in the plateau it does not (any full-rank, full-scale implant un-sticks a primed host; structure adds speed but is no longer necessary). What the grokked donor carries that random does not: a post-cleanup relevant-bit share of 0.83 against the 0.075 chance level (squared-power convention; Appendix D.5) — present even though it barely leaves chance during the host's own plateau.

B.11 The synthetic-embedding control and the 8-pair cross-run battery

Synthetic embeddings (the decisive class control). Untrained, analytically specified cosine banks — six frequencies, four dimensions each, random phases per input position, scaled to the donor's norm — implanted into the seed-0 host @30k over a 20k horizon. On the donor's six frequencies: 0.844 @2k, 1.000 by 16k. On six *randomly chosen* frequencies: 0.828 @2k, 0.996 @20k. Both rescue; random directions at the same norm fail or harm (B.5, B.9). Frequency identity and learned content are not the bottleneck: what suffices is class membership at the right scale. The decomposition battery (three further matched-norm implants, run against the registered prediction that only right-class waves rescue) settles which property of the bank is load-bearing:

Implant at donor norm (seed-0 host @30k)	@2k	$t_{0.9}$
Learned embedding	0.932	1.2k
Cosine bank, donor's 6 frequencies (position-shared)	0.844	2.9k
Cosine bank, 6 random frequencies (position-shared)	0.828	3.9k
Cosine bank, donor's frequencies **+0.125** (shared, off-lattice; registered)	0.752	4.25k
Cosine bank, donor's frequencies **+0.25** (shared, off-lattice; registered)	0.789	4.5k
Cosine bank, donor's frequencies **+0.375** (shared, off-lattice; registered)	0.716	5.5k
Cosine bank, donor's frequencies **+0.5** (shared, off-lattice; registered)	0.610	9.75k
Cosine bank, donor's frequencies **+0.75** (shared, off-lattice; registered)	0.759	4.0k
Cosine bank, donor's frequencies **+1.0** (shared, *another character*; registered)	0.817	3.0k

*One caveat on the +1.0 cell: shifting the donor's six frequencies by one lands on {25,16,36,49,31,26}, and **two of those six are donor characters** — 25 outright, and 49 by reflection, since 49 = 97 − 48 gives cos(2π·49t/97) = cos(2π·48t/97) identically on integer t and 48 is one of the donor's six. Four of six are disjoint. The cell is therefore a weaker "maximally perturbed" reference than it appears, and the periodicity reading rests on it: we flag this rather than rerun at a shift that avoids the donor set.*

Cosine bank, position-unshared frequency sets (single draw; superseded by the k = 0 cell of the rank-matched sweep, median 12,500)	0.496	∼13k
Random rank-6	0.412	∼11k
Random rank-6, position-**shared** (registered)	0.481	11.5k
Smooth-aperiodic (non-wrapping Gaussian-smoothed noise)	0.328	∼13k
Smoothed noise, position-shared (registered; 0.96 of its power in 6 low bins)	0.363	never (0.735 @20k)
No implant (fresh optimizer)	0.240	12.7k
Random direction	0.011	never (0.346 @20k)

One resolution note, owed here as it was in B.8: the three earliest failing arms are read off a 2,000-step evaluation grid where the baseline and every registered arm are on 250 steps, so their crossings are quoted to that resolution and no finer. Low rank alone, smoothness alone, periodicity alone and *sharing alone* all track the baseline, and no conjunction of them is sufficient either: the shared smoothed-noise bank carries 0.96 of its power in six bins — passing the same top-6 test that certifies the donor embedding at 0.71 — and never crosses. What the shared banks isolate instead is a *lattice-distance effect*, and sweeping the offset shows it is **periodic rather than decaying** — but a phase-draw null bounds how much of the sweep is readable. Redrawing the on-lattice cell's phases five times gives $t_{0.9}$ from 3,000 to 5,250, a 2,250-step spread against the 1,000 we registered (**falsified**), and that spread covers the eighth and quarter offsets outright (4.25k, 4.5k) and leaves the three-eighths cell (5.5k) just 250 steps beyond a five-draw maximum — outside the measured null, but by less than the gap between adjacent draws in it, so we treat none of the three as separable. One contrast survives it: the half-integer is slow at 9.75k against 2.9k on the lattice. A full-integer offset returns at 3.0k, which would suggest the effect has the period of the lattice rather than decaying with raw offset. That cell cannot carry the claim, and the dose-response says why: shifting the donor's six frequencies by one lands on {25,16,36,49,31,26}, of which 25 is a donor character outright and 49 is the reflection of donor character 48, and kill #41 shows that **two** shared frequencies already buy the full rescue (k = 2 crosses at a median 2,500). The +1.0 cell's speed is therefore accounted for by the two characters it accidentally retains, not by any periodicity in lattice distance. We therefore claim only the binary contrast — on the character lattice against a half-integer off it — and no statement about what governs the cells between. Exact characters are the optimum rather than the requirement; cross-position sharing is necessary throughout, every unshared bank tracking the baseline; and concentration at the wrong frequencies does not rescue at all, the shared smoothed bank never crossing. The registered prediction that off-lattice banks would not rescue is **falsified** by the quarter-step arm (kill #39); the prediction that none would reach the on-lattice band held — cosines of the form the task's characters take, which is why the (Chughtai et al., 2023) irrep grounding is structural rather than decorative. Within the sufficient class, the learned embedding retains a speed advantage over the best synthetic bank of ∼2.4× on the single draw of record ($t_{0.9}$ 1.2k vs. 2.9k) and 2.5–4.4× against the five phase draws, and the donor's own frequency set beats a random set only modestly (2.9k vs. 3.9k, both single draws). The registered character-theoretic prediction is confirmed and sharpened: it predicted waves would be necessary; the measurement adds that cross-position sharing is the necessary part of waveness.

The 8-pair battery. Eight population hosts at 30k, each paired with a different run's 150k final as donor, none/embed/rand-normed per pair, 2k fine-tune on the host's own split; per-pair rows, never averaged:

Host (checkpoint acc @30k)	Donor (final acc / norm / top-6 share)	none	embed	rand-normed
i0_d1 (0.913 — late transition)[2]	i2_d0 (0.988 / 39.50 / 0.912)	0.926	0.766	0.269
i0_d2 (0.158 — deep plateau)[2]	i3_d0 (0.922 / 42.69 / 0.544)	0.380	0.814	0.026
i1_d0 (0.142)	i5_d0 (0.989 / 32.24 / 0.777)	0.221	0.678	0.161
i2_d0 (0.169)	i8_d0 (0.975 / 45.19 / 0.844)	0.276	0.689	0.006
i3_d0 (0.117)	i9_d0 (0.748 / 29.85 / 0.457)	0.215	0.480	0.085
i5_d0 (0.077)	i0_d1 (0.9957 / 34.83 / 0.910)	0.111	0.634	0.209
i8_d0 (0.073)	i0_d2 (0.9509 / 38.84 / 0.668)	0.108	0.847	0.020
i9_d0 (0.082)	i1_d0 (0.936 / 37.82 / 0.548)	0.104	0.971	0.067

Three readings, each phase-resolved (Section 3.3): deep-plateau hosts rescue 0.480–0.971 (median 0.689) against baselines of 0.104–0.380; the post-grok host is harmed (embed 0.766 against none 0.926) — both boundary phases since replicated and complicated by the registered battery below.

The phase-replication and donor-age battery (registered before launch; host accuracy at load | none | embed, 2k fine-tunes on the host's own split):

Cell	Host (acc at load)	Donor	none	embed
post-grok	i6_d0 @30k (0.959; grokked @29k)	i0_d4 @150k	0.969	0.963
post-grok	i0_d3 @30k (0.980; grokked @28k)	i0_d6 @150k	0.987	**0.355**
mid-transition	i0_d4 @60k (0.417)	i0_d7 @150k	0.648	0.856
mid-transition	i0_d8 @66k (0.508, climbing; t50 66k)	i0_d9 @150k	0.600	**0.859**
donor-age (100k donor)	i9_d0 @30k (0.082)	i1_d0 @**100k**	0.104	0.846

[2] These two rows are re-measured. `modadd` builds its train/test split from the seed it is given and `grok.py` trains each run on its **data_seed**, but the battery script built the evaluation split from `seed`, which is 0 for every population run. The six hosts with data_seed = 0 are therefore unaffected; i0_d1 (data_seed 1) and i0_d2 (data_seed 2) had been scored on a split overlapping their own training data. Corrected values are above; the contaminated readings were 0.954 / 0.989 / 0.700 / 0.298 and 0.527 / 0.773 / 0.777 / 0.004. We found this while checking why two host accuracies disagreed with A.1; the obvious explanation, a run-name conflation, was not the cause.

[1] An earlier draft excused this cell as mislabeled, on the grounds that i0_d4 is A.4's non-monotone run. That is wrong: A.4's non-monotone run is **i4_d0**, and i0_d4 is a clean grokker with t50 = 60.5k, so its 0.417 at 60k is exactly a genuine climbing mid-transition host. The excuse is withdrawn and the cell counts (kill #27).

The post-grok registered prediction (harm) is confirmed in sign on both cells with wildly different magnitude (-0.006 within epsilon; -0.632 catastrophic — with the corrected i0_d1 cell's -0.160, the phase is 3-for-3 at-or-below none). The mid-transition registered prediction (no gap) **failed on the clean cell** — i0_d8, genuinely climbing at 0.508, is helped substantially (kill #27) — so the middle phase failed 2 of 2 once i0_d2 is re-scored on its own split — at 0.158 it is a deep-plateau host, not a mid-transition one, and its embed-over-none gap is +0.434 rather than the contaminated +0.004 — and Section 3.3 claims only the two robust phases. **A host selected for a property it turned out not to have (registered; kill #40, withdrawn).** Section 3.3's law and Section 3.2's class result together predict that a host which has *completed* its cross-position sharing should not be much helped. We selected i0_d1 @15k on D.3's record of a six-of-six crossing at **step 700** — it does not grok until 36k, and at 15k it reads **0.138** on held-out data — and measured a rescue: implanting its own 150k embedding at the host's norm crosses at $t_{0.9}$ **2,500 against a 9,500 baseline**, a $3.8\times$ rescue where the registration required $\geq$ 4,750. On then checking the host's own checkpoints we found it at **five-of-six, not six-of-six**, at 15k — and at five-of-six at its fully grokked 150k checkpoint too. The property is not latched by the first crossing but flickers throughout training (D.3), so the cell does not test its registered claim and **we withdraw it**. The measurements stand as a dose-response point — a five-of-six deep-plateau host, rescued $3.8\times$ — and carry no falsification. The analytic bank at the same norm reads 6,000 (the registered $2\times$ band, missed at $2.4\times$); a random direction at the same norm reaches 0.003 at 2k against the baseline's 0.335 and never crosses. Whether a host with *durable* six-of-six sharing is rescued is untested, and given the flicker such a host may not exist in this population.

Does the implant supply what is missing, or amplify what is there? (registered; kill #42). Everything above is consistent with (Varma et al., 2023): the donor embedding might not be supplying structure the host lacks so much as amplifying structure the host already partly holds. The separating control decomposes the donor embedding against the *host's own* 30k row space and implants each part at donor norm — the component lying inside that span, and the component orthogonal to it. We registered that the orthogonal arm would rescue ($t_{0.9} \leq$ 4k) and the in-span arm would not ($\geq$ 8k), because on our reading what the host lacks is what matters; falsified if the in-span arm crossed first. **Both registered clauses fail.** The in-span arm crosses at **4,250** and the orthogonal arm also at **4,250**, against a 12,713 baseline — and at the 2k read the in-span arm is far ahead, **0.823 against 0.286**. The tie turned out to be the grid rather than the network. **Retested at 25-step resolution on three hosts** (29k/30k/31k, B.8's host-shift convention), the arms separate: in-span 3,750 against orthogonal 3,875 at 29k, 4,375 against 4,225 at 30k, and 3,600 against 3,675 at 31k, with the no-implant arm crossing on none of the three within 8k. The in-span arm crosses first on two hosts of three, so the falsification criterion is met on its own terms and not by our reading of the bands. We attach no weight to the ordering itself: the margins are 75–150 steps, the sign flips between hosts, and re-running the 30k cell under a different thread count moves a crossing by a grid step, so the two arms are not separable on $t_{0.9}$. What *is* stable is the early read — the in-span arm sits at 0.834, 0.823 and 0.830 at 2k across the three hosts while the orthogonal arm reads 0.458, 0.289 and 0.527 — so the component of the donor already inside the host's span is not merely as good as the part outside it but reliably better early. What the host already spans is not inert: the projection of the donor into the host's own row space rescues as fast as the part outside it, and faster early. That is evidence for the amplification reading over the supply reading, and it is the clearest result we have that runs against our own framing. **The geometry of that decomposition, measured.** The donor embedding carries **87.1%** of its Frobenius *norm* inside the host's 30k row space (‖P_H D‖/‖D‖ = 0.8710, i.e. 75.9% of its energy; ‖H‖ = 74.13, ‖D‖ = 43.66). Two nulls place that figure: an isotropic random 24-dimensional subspace catches 0.350 ± 0.025 over 200 draws, and 19 independently seeded foreign 30k hosts catch 0.659 ± 0.025, so the same-lineage host sits 8.4 foreign-host standard deviations above a baseline that is itself well above chance. But the in-span part is not a rescaling of the host: its cosine to the host embedding is 0.529, the best scalar fit leaves a relative residual of 0.848, so **72% of the in-span energy is orthogonal to the host inside the host's own span**, and writing P_H D = A H gives an A that is 92% off-diagonal — a recombination, not a per-row gain. The failing control of B.5 is the pure rescaling (cosine 1.000, residual 0) and reaches 0.533. One asymmetry we cannot remove: at 87% energy the in-span arm rescaled to donor norm is close to the donor embedding itself, while the orthogonal arm is a residual carrying 49.1% of the donor's norm, amplified 2.04× to match it, so the two arms are not matched in how far each sits from a trained embedding. The static geometry is measured; the causal attribution of the 2k gap to the residual is a joint reading of two measurements, not a third experiment.

An isogenic post-grok host (registered). Section 3.3's harm cells all take the donor from a *different* run, and independently seeded runs converge to largely disjoint frequency pairs (C.3). A cross-run embedding dropped into a host whose readout is already committed to another basis would scramble the circuit for a reason that has nothing to do with maturity, so the harm could be basis incompatibility rather than phase. The separating control is isogenic: a run's own 150k embedding, rescaled to the host's norm, implanted into that same run's own post-grok checkpoint. We registered that such cells would still land at or below their own no-implant baseline, falsified if a majority landed clearly above it. It was scored **confirmed, three of three** — i0_d3 at 48k reads 0.9894 against 0.9943, i0_d1 at 56k 0.7545 against 0.9826, i6_d0 at 49k 0.9758 against 1.0000 — and **we withdraw that scoring**. The registration could not have failed. Falsification required a majority of cells to land more than 0.02 *above* baselines of 0.9943, 0.9826 and 1.0000, and accuracy caps at 1.0, so the headroom available was 0.0057, 0.0174 and 0.0000: zero of three cells could have fired it. The defect is structural rather than unlucky — hosts are selected 20k steps past their grok, which guarantees a near-ceiling baseline — and the repair is a threshold stated as a fraction of the headroom that exists rather than as an absolute gap. Read past 2k the cells reverse as well: the implant first resets the host to chance (0.0094 to 0.0164 against 1/97 = 0.0103), and by 12k all three are back at or above their own baseline (+0.0017, +0.0121, +0.0021), with i0_d1 passing **+0.0213 at 10k** — above the very threshold the registration could not reach at 2k. So it is unfalsifiable at the horizon it was scored at and falsified at a longer one. It establishes neither the direction nor the magnitude it was cited for, and Section 3.3 no longer rests on it (`runs/isogenic_horizon.jsonl`). The earlier

printed cells (0.987/0.995, 0.810/0.982, 0.976/1.000) predate the seed/data_seed repair of footnote 2 and are superseded by the values above.

The same horizon on the cross-run harm cells. The post-grok cells above are read at 2k like the rest of the battery. Run to 12,000 steps they behave as the isogenic cells do: the implant first resets the host to chance (zero-shot 0.0125 to 0.0164 against 1/97 = 0.0103), and the deficit then closes.

Post-grok host @30k	Donor	zero-shot	gap @2k	gap @12k
i6_d0 (0.9592)	i0_d4	0.0132	−0.0043	**+0.0004**
i0_d1 (0.9126)	i2_d0	0.0164	−0.2142	**+0.0017**
i0_d3 (0.9802)	i0_d6	0.0125	−0.6327	**−0.0601**

Two of the three return to at or above their own baseline by 12k and one keeps −0.0601, which is the reading Section 3.3 cites (`runs/crossrun_horizon.jsonl`). The i0_d1 row reads −0.2142 at 2k here against the −0.160 printed above: it is a boundary-phase cell and the two values are the thread-count envelope B.2 documents, not a disagreement about the cell. The direction of the phase law is unchanged at either horizon. What the longer read removes is any claim that the cost is damage rather than repair — on two of three cells it is fully transient, and on the third a 0.633 deficit at 2k is a 0.060 deficit at 12k.

Sharing at matched rank and matched frequency count (registered). The necessity claim above rested on a single draw of one cell, and that cell was not rank-matched: a cosine bank spans a cosine and a sine at each frequency, so each block is rank 12 where the random rank-6 control is rank 6, and its unshared arm used twelve distinct frequencies against the shared arm's six — confounding sharing with frequency count. We therefore swept sharing at **fixed rank and fixed per-position frequency count**: each block always carries exactly six frequencies, k of them common to both, for k = 0, 2, 4 and 6, three phase draws per cell. Median $t_{0.9}$ reads **12,500 at k = 0, 2,500 at k = 2, 3,250 at k = 4 and 3,250 at k = 6**, against the 12,713 baseline. Two registered clauses hold: k = 0 tracks baseline ($\geq$ 10k on all three draws) and k = 6 reaches the on-lattice band ($\leq$ 4k on all three). **The monotonicity clause fails.** Sharing is not a dose. The entire effect is a step between zero and two shared frequencies, and k = 2 is if anything the fastest cell; k = 4 and k = 6 are indistinguishable from it and from each other against the phase spread measured above. What survives this cut is the necessity claim, now unconfounded by rank and by frequency count — an unshared bank is five times slower than any shared one — and what we withdraw is any reading on which more sharing is faster.

An inert donor, and what it is missing (registered). The batteries above take every donor at grok or at the final checkpoint, so none of them separates *the plateau waits on the mature representation* from the weaker *any coherent foreign representation at the right scale dislodges the memorized solution*. We ran the separating control on one foreign lineage at two ages, with the donor rescaled to the flagship donor's embedding norm in both arms so that only maturity differs. i0_d6 at 30k is deep pre-grok — test 0.0725, and that run's own t50 is 98k, so it has no solution to donate — yet its embedding is not spectrally empty (top-6 share 0.614 against the mature checkpoint's 0.875). Implanted into the seed-0 host at 30k it **never reaches 0.5 in 20,000 steps**, finishing at 0.249: not slow but inert, and below the structureless random-direction control's 0.346 at the same horizon (B.9). The same lineage's 150k checkpoint, rescaled identically, crosses 0.9 at **2,250** — faster than any synthetic bank. Both halves were registered before the run. The reading is not that maturity as such is required — an *untrained* random-frequency bank rescues at 3.9k, with no maturity at all. It is that this donor is not yet in the sufficient class. Its two input blocks' top-6 frequency sets agree on only 4 of 6 at 30k ([11,15,25,29,36,41] against [11,25,27,29,36,47]) and on 6 of 6 at 150k ([13,25,26,29,36,39] in both), so the pre-grok checkpoint is short of full agreement where the mature one is not. **An earlier draft read that as the explanation** — four-of-six being, it assumed, the baseline-tracking cell Section 3.2 identifies — and called it the sharpest confirmation of the class theory we had. The rank-matched dose-response above refutes it: a four-of-six shared bank crosses at a median 3,250, comfortably inside the sufficient class, and the baseline-tracking cell is k = 0, not k = 4. We withdraw the explanation and the claim built on it. The measurement stands and is unexplained: a spectrally non-empty pre-grok learned embedding, at donor norm, is *inert* where an untrained random-frequency bank rescues at 3.9k. Sharing count does not account for it. We record it as the clearest open anomaly in this battery rather than supply a second post-hoc reading for it.

The donor-age pair: the battery's strongest rescue drops from 0.971 (150k donor) to 0.846 with the *same* donor taken at 100k — age moves rescue quality within the rescue range, not the phenomenon. Donor consolidation shows a **floor effect, not a predictor** (kill #23): the one unconsolidated donor (top-6 share 0.457) gives the weakest rescue (0.480), but above that floor concentration does not predict quality — share 0.548 delivers the battery's best rescue (0.971) while share 0.910 delivers 0.634; host-side variation dominates. One uncontrolled difference is flagged: these donors are taken at 150k against the flagship donor's 100k, so donor age may contribute to cross-run rows. (Bookkeeping: the donor "final acc" column inherited the same `seed`-versus-`data_seed` split bug as footnote 2's host rows. Re-scored on each donor's own split, i0_d1 reads 0.9957 and i0_d2 0.9509 at last eval (0.9957 and 0.9471 as means of the last five), so the printed 0.998 and 0.972 are contaminated readings and not a convention difference: no convention on the corrected split produces them. An earlier draft explained the gap as a convention difference; that explanation was wrong and is withdrawn.) The registered prediction ("median embed rescue > 0.9 across pairs") was optimistic — the plateau-host median is 0.68 — and the miss is reported, not reframed.

B.12 Registered-prediction bookkeeping

On multiplicity. This ledger indexes the registered predictions behind every battery in this paper; the released log carries the full record, whose kill entries run to #42 — narrated contemporaneously through #38, and from #39 onward supplied as an explicitly labelled reconstruction from registrations committed before each run (Reproducibility statement). Of the 28 registrations indexed below, 19 are recorded as falsifications, and several of the paper's load-bearing claims reached their present form by being falsified first. Pre-registration controls selective reporting; it does not control the family-wise error rate, and we apply no correction. Readers should calibrate accordingly: a single cell confirming its prediction at n = 1 is weak evidence, and the claims we rest weight on — the rebuild-time ordering, the class decomposition, the phase law — are those confirmed across multiple cells or by more than one instrument. The falsifications are the more informative half of this table.

Every battery cell above was run against a prediction registered before the outcome — in the research log through kill #38, and from #39 onward in the docstring of the script that runs it, committed to version control before the run it could be falsified by — and the falsifications are part of the record. (Kill numbers are inherited from a longer-running log and the sequence is **not dense**: #2–#7 and #14 were never assigned and no entries exist for them. Of the rest, #8–#13, #15–#17 and #20 concern earlier instruments and the companion program and are in the released log; no entry was ever assigned to #1 either. #30 is the parity probe-sensitivity control and is reported in Section 1 and D.5 rather than as a bullet here. #21 is the one entry described in this paper whose outcome is logged contemporaneously but whose prediction was never registered in advance — see the Reproducibility statement.)

- **Kill #18.** The radial hypothesis for deep-plateau parity ("any embedding shrink un-sticks the host") registered "random at host norm → dead"; it rescued (0.859 @4k → 1.000). The radial task-axis reading died with it.
- **Kill #19.** The "anesthetic consolidated donor" reading of the deep-host donor failure died when donor-at-host-norm rescued fastest of the nine deep-host cells (1.000 @4k): the failure was scale starvation, not consolidation toxicity.
- **Kill #21.** The registered prediction that the 60k donor@1.3 failure would replicate across neighboring hosts was falsified (rescues at 50k and 70k); the 0.479 cell is a one-checkpoint outlier, reported as such.
- **Kill #24.** Kill #22's 200-step circuit-identity effect, real against draw spread, was itself withdrawn when the registered host-shift bootstrap measured a 300–400-step within-condition spread across hosts 29k/30k/31k — the paired within-host lead (200–300 steps, 3/3 hosts) does not exceed the honest error bar (B.3). **This withdrawal was itself wrong in kind and is withdrawn by kill #31 below**, which redid the test paired across eight hosts; both stages stay in the record.
- **Kill #23.** The donor-concentration-predicts-quality reading, written from the 8-pair battery's first five rows, was killed by its final row (share 0.548, rescue 0.971): above the consolidation floor, concentration predicts nothing (B.11).

- **Withdrawn: the isogenic post-grok control (B.11).** Before running it we registered that a run's own late embedding, implanted into its own post-grok checkpoint, would still land at or below that host's no-implant baseline — falsified if a majority of cells landed clearly above. We scored it confirmed three of three, and that scoring is withdrawn: against baselines of 0.9943, 0.9826 and 1.0000 with accuracy capped at 1.0, a falsifier demanding more than +0.02 above baseline had 0.0057, 0.0174 and 0.0000 of headroom, so no cell could have fired it. Read to 12k all three return to at or above baseline and one passes +0.0213, so the registration is unfalsifiable at the horizon it was scored at and falsified at a longer one. The confound it was built to address — that Section 3.3's harm cells are all cross-run, so a foreign basis could harm a committed host for reasons unrelated to phase — is real and remains unaddressed. This entry is the ledger failing at its own job, and it is recorded rather than quietly dropped.
- **Kill #42.** Ours, registered before the run. Every result in this paper is consistent with the implant *amplifying* structure the host already partly holds rather than supplying what it lacks (Varma et al., 2023), so we decomposed the donor embedding against the host's own 30k row space and implanted each part at donor norm. We predicted the orthogonal component would rescue ($t_{0.9} \leq$ 4k) and the in-span component would not ($\geq$ 8k). **Both clauses fail**: both arms cross at 4,250 against a 12,713 baseline, and at 2k the in-span arm leads 0.823 to 0.286. The falsification criterion as written — in-span crossing first — looked unmet because the two arms tied on a 250-step grid. A registered retest at 25-step resolution across three hosts breaks the tie and the in-span arm crosses first on two of the three, so the criterion is met as written. The crossing margins are 75–150 steps and the sign flips between hosts, so we rest the conclusion on the 2k reads instead, which are stable: in-span 0.834/0.823/0.830 against orthogonal 0.458/0.289/0.527. The amplification reading is supported and our supply reading is not (B.11).
- **Registered, confirmed: the frozen complement (B.7).** The mirror of B.7's frozen-embedding arm — donor wiring and head frozen, only the embedding trainable. Registered at $t_{0.9} >$ 2k, falsified at or before 2k. It reads 0.037 at 2k and never crosses 0.9 within 20,000 steps, against B.7's 0.763 at 2k in the opposite configuration.
- **Kill #41.** Ours, registered before the rank-matched sharing sweep and stated in three clauses. Two held: an unshared bank tracks baseline (k = 0 at $\geq$ 10k on all three draws) and a fully shared one reaches the on-lattice band (k = 6 at $\leq$ 4k on all three). **The monotonicity clause failed.** $t_{0.9}$ does not fall with k; it steps down between k = 0 and k = 2 and is flat thereafter, with k = 2 the fastest cell. Neither falsification threshold was crossed, so the necessity claim the sweep was built to secure survives — now unconfounded by rank and by frequency count, which the single-draw cell it replaces was not — while "more sharing is faster" does not. The registration also named k = 4 as the cell that would decide B.11's inert foreign donor: under a graded law that donor is a counterexample, under a cliff it is explained. The answer was neither, because the cliff sits at k = 2 rather than at k = 6. Four-of-six rescues, so **the donor remains a counterexample**, and we withdraw the sharing-count explanation of it together with the sentence that called it the sharpest confirmation of the class theory we had. The anomaly is reported unexplained in B.11.
- **Kill #40 (withdrawn).** Ours, registered before the run with the expectation stated that it would fail. Section 3.3's law plus Section 3.2's class result predict that a host which has already completed its cross-position sharing is not much helped; we required $t_{0.9} \geq$ half the baseline and measured a 3.8$\times$ rescue on i0_d1 @15k, selected on D.3's record of a step-700 crossing. **The registration's precondition did not hold on the artifact we used:** the host reads five-of-six at 15k, not six-of-six, and the property flickers for the whole run (D.3). The cell does not test its claim and we withdraw it rather than count it. We keep it in the ledger for what it does establish, which is methodological: a registration must verify its own precondition on the specific artifact it will use, not inherit it from a summary table. The sharing rung is retired as a training-side anchor on two findings independent of this cell — that it orders no better than the pooled-power rung it was introduced to replace, and that it flickers rather than latching (both D.3); Section 1's exploitability framing rests on those and on Appendix E, none of which depend on this cell. A companion registration of ours also failed: we predicted the layer-0 wiring would learn to *read* the support-dedicated embedding row after building it, and it does not — the read fraction of that row is 0.010–0.021 at its appearance against a uniform 0.040, and still at or below uniform at the grok

on all four filmed lineages. The row-wise signal of D.5 therefore stands as a predictor and not as a mechanism.

- **Kill #39.** Ours, registered before the off-lattice synthetic arms. We predicted that cosine banks at non-integer frequencies would not rescue (falsification at $t_{0.9} \leq$ 6k); the quarter-step arm crosses at **4.5k** and the prediction is falsified. The companion clause — that no off-lattice arm would reach the on-lattice band — held. What replaced it is weaker than an earlier draft claimed: rescue speed recovers at a full-integer offset, so exact characters are the optimum rather than a boundary, but a five-draw phase null on the on-lattice cell spans 3,000–5,250 and covers the eighth and quarter offsets outright and falls 250 steps short of the three-eighths cell, leaving only the half-integer contrast readable (B.11). The graded reading of this sweep is therefore withdrawn, and we do not carry the periodicity into the abstract.
- **Registered, confirmed: the row-wise plateau signal (D.5).** Before training three further parity lineages under A.5's procedure we registered that (P1) any new lineage failing to grok within 150k would also never sustain max-row support purity ≥ 0.50, and (P2) any new plateau-then-grok lineage would sustain it strictly before its grok. Both held: seeds 15 and 17 ran the full budget at chance with maxima 0.220 and 0.305, and seed 10 crossed at 7.5k and grokked at 44.5k. Four further seeds (11, 12, 13, 14) generalized within 6.5k steps with no plateau and were terminated early as uninformative, which the registration specifies in advance; their traces are retained. A fifth, seed 16, ran the full budget and groks at 11.5k with no plateau; it is outside both clauses and is reported in D.5 rather than counted here. With it the eight no-plateau lineages of Section 3.5 are seeds 2, 3, 5, 11, 12, 13, 14 and 16, whose grok steps span 0.5k (seed 3) to 11.5k (seed 16).
- **Kill #38.** Ours, registered before running the parity battery on the third and fourth plateau lineages. Two of the three predictions failed. The scale-matched embedding was registered at $t_{0.9} \leq$ 0.5k on both and reads 1.22k on i7. The norm-matched random implant was registered to sit at or below the no-implant arm at 20k and sits marginally above it on i4 (0.993 against 0.984), although it is twice as slow to cross 0.9. The second failure is worth recording for a second reason: we first scored that cell a pass by comparing crossing times rather than the 20k accuracy the registration named, which is the error of grading a prediction against the statistic that happens to favour it. The main prediction, P1, was confirmed on both lineages and is what the section rests on.
- **Kill #37.** Ours, three mechanisms proposed while filming all four plateau lineages, explored on two and tested on two held out. (i) That the embedding never changes and the *wiring* learns which of its existing dimensions already carry support-bit power — false: that read-weighted share tracks the global share almost exactly and sits at chance through the plateau, so the wiring is not selecting for support-bit power. (ii) A threshold law, that layer-0 match is ≥ 0.40 when accuracy leaves chance — falsified on one holdout lineage at 0.391. (iii) An ordering law, that layer 0 leads layer 1 throughout — falsified on the same lineage, where layer 1 is briefly ahead. The descriptive facts behind (ii) and (iii) survive as ranges and are reported that way in D.5. Also recorded dead so they are not re-proposed: the layer-0 chooser softmax entropy is flat through the plateau — the film's `ent0` column is a different quantity, declines on all five lineages, and separates nothing — and the four films do not collapse onto a universal curve under rescaling by plateau length.
- **Kill #36.** Ours, registered before filming the eighth parity lineage, which memorizes and never groks. We predicted that layer-0 wiring stability would separate a run that groks from one that does not — that a plateau which never ends would be one whose wiring never settles. It does not separate them. The registered criterion in fact **passes**, by four percent; we report the pass and conclude against it, because its baseline was post-grok churn — cleanup rewiring rather than settling — and because two grokking lineages individually churn more after grokking than the never-grokking one does late in its run. The negative rests on an independent coarser measure, lag self-match, on which the never-grokking lineage sits inside the grokking range at both windows tested (D.5). Nothing we can measure *during* a plateau tells the eighth lineage apart from the four that grok: its embedding is at chance like theirs and its wiring churns like theirs. The inversion therefore survives as a retrospective claim — on runs that plateau and then grok, the progress the plateau hides is in the wiring — and not as a predictive one. This is the third plateau-side predictor we have proposed and failed

to confirm, after kill #33's step-500 separator — refused by a screen at a different split, and so scoped to that configuration — and kill #34's frequency balance; we record the pattern, because a reader should weigh it against any prediction of this shape, including ours.

- **Kill #35.** Ours, registered before filming parity lineage i4. D.5 reports a norm arc on both earlier lineages — the embedding norm peaks mid-plateau (both at ∼40k) and erodes through the plateau's entire second half, an Omnigrok-style radial contraction running tens of thousands of steps before the snap. We registered that i4's norm would peak between 20k and 60k. It does not: i4's norm is double-humped, reaching 16.0 at 40k, falling to 11.6 by 60k, rising again to its global pre-grok peak of 17.9 at 100k, and only then eroding to 10.3 at the grok. The erosion clause survives (0.58 of peak at grok, against a registered $<$ 0.60) but the timing does not, so **the mid-plateau peak is a property of two lineages and not of the task**. What is left is the weaker and still useful statement that the norm contracts across the approach on all three, which is what B.10's scale-matching law actually needs.
- **Kill #34.** Ours, proposed while revising. The four prints show every network needs two frequency families, and on four runs the balance between them rose monotonically through the plateau — suggesting the plateau is spent bringing the second family up, and that the network groks as they approach parity. At population scale it fails twice. Computed online, as the ratio of the second-strongest to the strongest frequency at each checkpoint, it already exceeds 0.6 at step 100 in 17 of 19 runs, because a random embedding has a near-flat spectrum; and the two censored runs reach 0.63 and 0.95, inside the grokking range. Computed with the final print's families and tracked backwards, it rises in 15 of 17 grokking runs **and in 2 of 2 runs that never grok**, with its value at the grok spanning 0.097 to 0.989 — i7_d0 groks with its second family at 0.097. The balance describes what training builds, not what the plateau waits for, and every timing, predictive and explanatory reading of it is withdrawn. The ablation result of C.2 stands.
- **Kill #33.** The registered parity-mechanism prediction — that the embedding's relevant-bit share at step 500 separates runs that generalise early from runs that stall, at $\geq 2\times$ chance and $\leq 1.5\times$ chance respectively — was **falsified on both clauses** by a six-seed screen: seed 6 generalized with its embedding at $1.1\times$ chance, seed 7 stalled at $1.8\times$ (D.5). The mechanism was ours, proposed from two lineages and killed by six. A note for anyone comparing this screen with Section 3.5: the screen trained on the full split where the population runs hold out a tenth for validation, so its runs are not the same runs. Seed 7 is unresolved in the screen and, trained under the population procedure, groks at 63k. The two statements are about two different trainings of one seed, and neither is a correction of the other. Because the screen is a different training, this kill retires the mechanism for the screen's configuration; it has not been retested under the population procedure, and we scope it that way rather than saying the population refused it. A post-hoc threshold at step 1,000 does separate the groups and is recorded as a hypothesis, not a finding.
- **Kill #32.** The registered replication prediction "both new parity lineages grok within the 150k budget" **failed on both**: seeds 2 and 3 generalize at ∼2k and ∼0.5k with no plateau at all. The inversion is therefore scoped in Section 3.5 and the abstract to runs that plateau. The ten-seed base rate this kill originally quoted was later withdrawn as well — it mixed a full-split screen with held-out-validation training runs — and Section 3.5 now reports eight lineages trained under one procedure. This cost a claim and was found by running the replication rather than by review.
- **Kill #31.** The registered eight-host paired redo of B.3 falsified its own primary prediction: the donor core leads randomly initialized replacements on 8 of 8 hosts (median 221 steps, sign and Wilcoxon p = 0.008), clearing the pre-fixed $\geq$ 7-of-8 and $\geq$ 150-step falsification threshold. **Kill #24's withdrawal is therefore itself withdrawn** — a kill of a kill. The original withdrawal was not conservative but wrong in kind: it tested a paired effect against an unpaired spread. The claim of record gains a magnitude (about 1.9% of the embedding's acceleration) where it previously had a zero. Two subsidiary readings asserted from the three-host data also died: the lead is a constant share of rebuild time rather than a declining dose-response, and random replacement units are near-neutral rather than harmful.
- **Kill #22 (two-stage).** The registered stopwatch prediction ("no separation beyond draw spread" between donor-core and random-core given the embedding) was falsified at within-host resolution — a consistent 200–300-step lead against a ˜zero draw-spread floor — and

then effectively restored when the registered host-shift bootstrap measured the honest error bar at 300–400 steps (kill #24): the endpoint null stands, its falsification an artifact of the wrong variance source (B.3). Both stages are part of the record — and a third stage now closes it, see kill #31.

- The symmetric battery's registration ("genuine uncertainty, prior leans rescue") resolved toward rescue with the ranking preserved (B.8); the overnight extension and cross-host runs were registered as consistency checks of the scale story, and behaved as predicted.
- The decomposition battery's registration ("only right-class waves rescue; if rank-6 suffices the claim weakens") was CONFIRMED and sharpened: rank-6, smooth, and position-unshared all track baseline; shared-frequency banks rescue (B.11).
- The transformer battery's registration (B.9) split: structure **confirmed** emphatically ($t_{0.9}$ $\approx$ 0.2k); **kill #28** — the norm-control prediction ("at or below baseline") falsified at the 2k read: in this architecture, and under its 10$\times$ larger weight decay, the random implant is destructive early but eventually un-sticks the host; the claim is quoted as graded wherever the transformer row appears.
- **Kill #29, in two stages.** The second-lineage parity registration ("donor wiring rescues the ripe host at $t_{0.9} \leq$ 4k") was falsified outright — with the donor taken at 150k the wiring arm ran 1.7$\times$ *slower* than no implant. The registered candidate mechanism (this lineage's film shows post-grok cleanup rewiring layer-0) was logged as a new prediction and confirmed: at-grok donor wiring rescues at 1.1k against a 5.6k baseline, and the two lineages' donors retain 0.495 and 0.021 of their at-grok programs respectively. The claim is restored with a condition it did not previously carry, and the falsification is part of the record (B.10).
- The capacity-matched head arm was registered before launch with a predicted $t_{0.9}$ of 6–10k and a falsification threshold at 3k (below which the dissociation would have been "dense maps are slow" rather than embedding-specific, and the abstract would have been rescoped the same day). The crossing was then re-measured on a 250-step grid, because the original evaluation grid jumped 2k $\rightarrow$ 4k and the 3k threshold lay inside that unmeasured interval: the head arm reads 0.897 at 3,750 steps and 0.915 at 4,000, giving a **measured $t_{0.9}$ of 3.79k**. That **misses the predicted band** — the head is faster than we guessed — while clearing the falsification threshold by a measured margin rather than an interpolated one; the embedding-specificity claim stands and the prediction miss is recorded. The registered early-harm sub-prediction was confirmed (0.147 at step 100, below the 0.208 baseline). Determinism check on the paired no-implant arm reproduced the values of record (0.240 at 2k; $t_{0.9}$ 12.9k on the 100-step grid this check used, against the 12,713 interpolated value quoted elsewhere — the same crossing read at two resolutions, not two measurements).
- The phase-replication battery's registration (B.11) resolved two-and-a-half ways: post-grok harm **confirmed** in sign on both new cells (magnitudes -0.006 and -0.632 — the law fixes the sign, the host fixes the size); **kill #27** — the registered mid-transition no-gap prediction resolved **2 of 2 against us**: it failed on i0_d8 (0.859 against 0.600) and i0_d4 (0.856 against 0.648). An apparent third cell, i0_d2, was counted as mid-transition on a host accuracy of 0.527 that B.11 footnote 2 retracts as contaminated; on its own split the host reads 0.158, which is a deep-plateau host, and it rescues 0.814 against 0.380. It is therefore not a test of this prediction and is not counted, and the 0.777 / 0.773 pair an earlier draft reported here is withdrawn with it. The middle phase therefore failed on both clean cells, which is what B.11 says and what an intermediate draft of this section briefly and wrongly denied after over-correcting a different error in the same cell. An earlier draft excused the i0_d4 cell as "mislabeled" on the grounds that it was A.4's non-monotone run; that run is **i4_d0**, and i0_d4 is a clean grokker whose t50 is 60.5k, so its 0.417 reading at 60k is exactly a genuine climbing mid-transition host. The excuse is withdrawn and the prediction is recorded as failing 2 of 2 — which makes the phase law cleaner, not weaker. This is the second i0_dN/iN_d0 conflation in this record (see A.1 footnote 1) and both are reported.
- The ripe-host converse battery's registration (B.10) resolved three ways, all reported: wiring-rescues **confirmed** ($t_{0.9}$ 1.7k $\leq$ the registered 4k); **kill #25** — "complement at least as fast as wiring-only" falsified (3k against 1.7k); **kill #26** — "embedding 2–4k, slower than wiring" falsified (instant, $\leq$0.25k; the prior 4k figure was a grid bound). The falsifications sharpened the claim's final shape (Section 3.5): implant exploitability is largely task-independent; what training grinds on is what inverts.

C ANATOMY ATLAS AND THE DENSE-MAP READINGS

C.1 THE CIRCUIT SKELETON AND ITS MOTIFS

Three units of the printed 193-line program, each shown in full and annotated with the measured commitment step of its wires (Section 4.1). Two are locked well before the grok; the third is still churning at 100k, which is the lock-in-versus-finality distinction in one view.

```
u0_9 = exp(clip(+0.60*e7 -0.58*1 -0.56))
       - ln(sp(+0.85*1 -0.87*e7 +0.84))
    # locked 56k
u0_0 = exp(clip(+0.66*e7 +0.66*e7 -0.55))
       - ln(sp(-1.01*e7 -1.01*e7 +0.99))
    # locked 53k
u1_17 = exp(clip(-0.53*e7 -0.77*u0_18 +0.49))
        - ln(sp(+0.06*u0_32 +0.27*u0_44 -0.09))
    # still churning @100k
 ... 190 more lines
```

The printed program's functional core resolves into 8 layer-0 units feeding 12 layer-1 units (a descriptive grouping from the formula read, distinct from B.1's threshold-defined k = 6 implant set — two criteria, both stated), and the motif vocabulary is multiplicative throughout: exp-of-sum structures compute products ($e^{\ln a+\ln b} = ab$), and cross-family combination is multiplicative via high-gain gating — u1_8 is a gate on e7 multiplying a 24-wave — not additive. The atlas's recurring specimens: an embedding *skip connection* (u1_19 reads coordinate e7 directly, bypassing layer 0); duplicated **twin units** u0_18 and u0_47 — functionally identical, with disjoint readers. Only u0_18 clears B.1's δ = 0.05 single-unit-ablation criterion, and the joint test that arbitrates load is the next measurement. On C.3's convention (all 9,409 inputs, baseline **0.9918**, recomputed for this revision): removing u0_18 costs 0.462, removing u0_47 costs 0.003, and removing both costs 0.729. What refutes an interchangeable backup pair is the **asymmetry**, not the joint cost — if either twin covered for the other, losing u0_18 alone would be cheap, and it is not. What the joint test adds is the converse: with u0_47 already gone, losing u0_18 costs 0.726 rather than 0.462, so u0_47 *does* buffer the loss of u0_18 — by 0.264, about a third of the 0.726 unbuffered cost — while contributing almost nothing on its own. That is a one-directional partial backup, and single-unit ablation cannot see it. One caveat we cannot rule out: ablation here is zeroing, and in a substrate whose units multiply an exponential branch by a logarithmic one, two simultaneous zeroed excursions may compound for reasons having nothing to do with redundancy. The asymmetry does not depend on that; the super-additivity might. The redundancy is architectural, not vestigial; and u0_44, the cleanest primitive in the program — exp(2.18·e11) — with 17 readers. Branch semantics are consistent across the program: critical layer-0 units put their non-e7 signals in the exp branch; critical layer-1 units live in the exp branch as anti-gates (of the form exp(−3.6u)); vestigial units are stock-gain e7/e7 in both branches.

C.2 THE E7 COMMITTEE

A committee of layer-0 units all read embedding coordinate e7, and the committee is rank-1: the second singular value carries 7.4% of the first's mass, and the members are amplitude-scaled copies of one ramp with a common kink (X-roots 0.40–0.445; Y-root standard deviation 0.016). It is *not* a threshold ladder — the members do not tile distinct input ranges — and criticality within the committee is extreme gain, nothing else.

The same anatomy across all four printed programs.

Appendix C otherwise reads one specimen. The structural claims were checked on all four hardened networks, evaluated at the hardening temperature T = 0.01 — loading a hardened checkpoint at the training temperature of 0.5 gives chance accuracy, a convention worth stating because it is easy to get wrong — and scored over all 9,409 inputs.

Network	acc	Large families (joint-ablation cost)	Tail
flagship	0.9918	f24×13 (0.881), f15×9 (0.987)	f35, f48, f27 $\leq$ 0.017
i0_d1	0.9982	f8×19 (0.910), f25×3 (0.928)	f4×2 (0.264)
i0_d5	0.9503	f1×16 (0.899), f9×4 (0.710)	f40, f27, f2 $\leq$ 0.031
i6_d0	0.9910	f19×11 (0.932), f28×10 (0.872)	f2, f40, f38 $\leq$ 0.080

Three things hold in all four. Every network rests on **exactly two large frequency families and needs both** — jointly ablating either one alone costs 0.71–0.99 accuracy, so no single family suffices. Every network carries a **tail of tiny families** whose joint ablation costs little, so f35's vestigial status is a recurring feature of these prints rather than a quirk of the flagship; i0_d1's f4 at 0.264 is the exception and is intermediate rather than vestigial. And **the frequencies differ every run while the structure does not**: the load-bearing pairs are {24,15}, {8,25}, {1,9} and {19,28}, a different pair each time, with no *low-order* harmonic relation within a pair — the closest is {1,9}, a ninth harmonic, and none is a 2:1 or 3:1 pairing.

That last point is the print-side counterpart of Section 3.2. The transplant shows any member of the character class suffices when supplied; the anatomy shows training picks a different member every run and arrives at the same architecture regardless. (The flagship's raw sort matches C.3's f24×13 / f15×9 / f35×2 once aliases are folded, f48 being 2×24 and f27 one of f35's aliased harmonics.)

We looked for a temporal claim alongside this and did not find one that survives; see kill #34.

C.3 The vestigial family and the distributed family

The final network's embedding dimensions sort into three frequency families: f24 × 13 dimensions, f15 × 9, f35 × 2 (dimensions e1 and e19). The f35 family — frequency 35 with aliased harmonics 27 and 8 — is carried entirely by non-critical layer-1 units (u1_9, u1_35, u1_33, u1_46; ∼9% of head mass) on a rank-1 phase axis (resultant 0.997), with redundant antipodal pairs that make it invisible to single-unit ablation. Joint ablation of the family — the test that closes the distributed-backup loophole — was reported as moving accuracy 0.9925 → 0.9918. That reading does not survive recomputation: 0.9918 *is* the unablated accuracy on this convention, so the pair cannot be a baseline and an ablated value. Remeasured from the released hardened checkpoint at T = 0.01 over all 9,409 inputs, zeroing e1 and e19 gives **0.9918 → 0.9896, a cost of 0.0022** — 21 of 9,409 inputs. (Appendix C scores ablations over all 9,409 inputs, not the test split: the flagship reads 0.9918 pooled against 0.9836 on test, both hardened.) Zeroing e1 *alone* costs more, 0.0174, which is the redundant antipodal pairing described above showing itself — removing one member breaks the cancellation, removing both is nearly free. The family's silhouette correlates with print errors at only +0.13: the pre-registered error-canceller hypothesis for f35 is dead, and f35 is genuinely vestigial (one caveat stands: the classification is by dominant frequency per dimension, and trace f35 content in mixed dimensions is not excluded). The f15 family is load-bearing but fully distributed: no single member matters, but jointly ablating its 9 embedding dimensions collapses accuracy from 0.9918 to **0.0049** (an earlier draft reported 0.133 here; recomputed from the released hardened checkpoint over all 9,409 inputs it does not reproduce, and C.2's cost of 0.987 for this family is the figure that does — erratum) — and the print-error bands sit at the f15 modes {15, 30, 45}: prints fail where the f15 family works, so the interference story belongs to f15, not f35. One surviving oddity, documented rather than claimed: e1:f35 *sharpens* between 50k and 100k under weight decay 0.1 while contributing nothing — why weight decay polishes a vestigial family is open. In the margin films, vestigial units end with *higher* final margins than core units (2.3 vs. 1.9) — persistence and confidence are not load.

C.4 The readout head, read in full

The head is a three-frequency matched-filter decoder, and each clause of that description is a measurement. 84.7% of its AC power lies in the 9 frequencies predicted by the wiring's fundamentals {15, 24, 35} and their mod-97 aliased harmonics. Its SVD needs 7 dimensions for 90% of variance — DC plus three families × two quadratures. Per-unit head columns are near-pure cosines (fit correlations up to r ≈ 0.95). The fitted phases cover both quadratures in the 15 and 24 families (phase resultants 0.39 and 0.34 — low resultants meaning spread phases, i.e., genuine quadrature coverage rather than a single phase axis). And antipodal push-pull pairs close the read: u1_14 against u1_19

correlate at -0.976. Between the wave-bank embedding (71% of non-DC power in 6 frequencies) and this matched filter sits the printed wiring; the dense parts are dense, and they are read.

D MARGIN FILMS PER SEED, THE 19-RUN LADDER TABLE, AND THE ELECTION/EROSION DIARIES

D.1 FILMS PER SEED

Seed 0 (flagship). The margin film's three acts, from raw checkpoints with zero training-time instrumentation: (I) the embedding's spectral structure is present from the first diaried checkpoint — power share in the final six frequencies 0.73 at step 1k (elected in the step-400–1000 competition, D.4), suppressed to $\sim$0.54 through memorization, recovering with generalization; an earlier reading that the frequency *set* is "in place by 15–25k" does not survive the raw-data check (under the DC-excluded convention the two input blocks' top-6 sets still differ into late training) and is retired; (II) the core layer-0 wiring locks essentially at the accuracy jump — median lock-in 54.5k against a grok time of 54k, with the twin units u0_18/u0_47 committing all four argument slots at 53k — and the core wiring's margin passes through its *minimum*, 0.37, at 30k: maximal indecision mid-plateau, not quiet consolidation; (III) layer-1 wiring locks at median 90k — 36k steps post-grok, in the logit-sharpening phase. Both act-II alignments are single-seed instances and are reported as such.

Seed 2. The ordering replicates — core L0 at 78k $<$ rest of L0 at 83.5k $<$ L1 at 88.5k — but the rungs commit *after* that run's t50 of 61k and *before* its grok time of 97.6k: one run, two timing statistics, 36k apart (the specimen quoted in Section 2). The rung-to-jump alignment is seed-contingent; the ordering is not, which is why the robust claim is the ordered ladder and "locks at the jump" is labeled an instance.

Seed 1, and the three films' rungs in full. The third full-resolution film (`emltree_wd0.1_s1`, grok 52.9k, t50 37.4k, 11 core units) was not previously tabulated; `margin_film_all.py` reads all three from raw checkpoints. Lock-in medians, core-L0 / rest-of-L0 / L1: seed 0 **53k / 53k / 90k**, seed 1 **89k / 89k / 90k**, seed 2 **78k / 83.5k / 88.5k**. Two facts follow, and the second is a limitation we had not stated. Layer 0 precedes layer 1 in all three, which is the ordering the main text claims. But **core-L0 and rest-of-L0 are tied in seed 0 and seed 1**, separating only in seed 2, so the intra-layer-0 rung split is a property of one film and not of the three; Section 4.1 now says so. Core margin minima are 0.37 at 25k (seed 0), 0.47 at 5k (seed 1) and 0.50 at 5k (seed 2), so the mid-plateau margin minimum quoted for the flagship is also seed-contingent.

The spectra rung sits on the diary floor. Recomputed on the commensurate convention — first crossing of halfway between the initial and final pooled share of power in each run's own final six frequencies — the spectra rung reads **1,000 in all three films**, which is the first diaried checkpoint. The reason is visible in the traces: that share *starts at or above* its final value (0.730$\rightarrow$0.710, 0.909$\rightarrow$0.835, 0.674$\rightarrow$0.610), because the six frequencies are identified at the last checkpoint and tracked backwards. The crossing is therefore a floor, not a measurement, and the full-resolution two-rung ordering is carried entirely by the wiring rungs' lateness. This is the concrete form of the caveat D.3 states in general terms, and it is why the commensurate population result, not this n = 3 one, is the claim of record.

The function probe. Against the worry that argmax identity is cosmetic while function drifts, a function-level probe patching locked wires was described in an earlier draft as confirming the commitments are functional. No statistic, host or n of record was retained for it, so we withdraw the claim from the main text and record it here as undocumented rather than restate it.

D.2 THE VERIFIED MARGIN TRACE

The flip-mechanism numbers are recomputed from checkpoints, with the definition attached: the **median layer-0 top-2 chooser logit margin** moves 0.469 at the 50k checkpoint $\rightarrow$ 1.462 at 60k — 3.1$\times$, "roughly trebles" — as layer 0 collapses to an e7-monoculture at 50–55k. The layer-1 median moves in the *opposite* direction over the same interval, 0.418 $\rightarrow$ 0.320, continuing downward: layer 1 is still churning while layer 0 commits, the ratchet of D.3 in margin form.

D.3 THE 19-RUN LADDER TABLE AND THE RATCHET

Under the strict last-change criterion (spectra $\leq$ layer-0 median $\leq$ layer-1 median — where the spectra rung's statistic is the step at which the winning family's power share crosses halfway from its initial to its final level, computed from the v2 diaries; the wiring rungs are per-wire last-argmax-change medians), the ordering holds in **14 of 19** runs; the 5 exceptions are budget-end saturations — both layers' last-change medians sit at 138k–147k against the 150k budget, wires still churning at cutoff, ordering unresolvable rather than violated. This is censoring language consistent with Section 2's treatment of the two censored runs.

Ladder verdict	Runs	Detail
Strict ordering holds	14	spectra $\leq$ L0 median $\leq$ L1 median under last-change
Budget-end saturation (censored)	5	both layer medians at 138k–147k vs. the 150k budget

The rungs, measured commensurately. The ordering above pairs a *threshold crossing* (spectra) against a *last-change time* (wiring). A halfway crossing is early by construction and a last-change time is late by construction, so some of the gap could be the statistics rather than the network. We therefore recomputed the layer-0 rung under the spectra rung's own convention: scoring each checkpoint's 192 layer-0 argument slots against the run's final program and taking the step at which that match crosses halfway from its initial to its final level. Under that commensurate pair the spectra rung precedes the wiring rung in **19 of 19** runs. The recomputed spectra statistic here is the pooled top-6 share, not the diary winning-family rung of the table above; the two agree on 18 of 19 runs and differ on i6_d0 (1.3k in the table, 12k recomputed). It lands at 500–900 steps in 17 of the 19, the other exception being i2_d0 at 61k, while the commensurate wiring crossing spans 25k–147k. The last-change statistic was not inflating the gap either: the two wiring times agree within a few thousand steps in most runs — 43k against 43k on i0_d3, 83k against 83k on i0_d4, 111k against 111k on i0_d2, 146k against 146k on i5_d0. `scratch_rung_commensurate.py` recomputes this from the saved checkpoints; no training is involved.

The sharing rung, per run. `scratch_sharing_rung.py` recomputes, for every population run and every saved checkpoint, the number of frequencies common to the two input blocks' top-6 sets. Full 6/6 agreement arrives at 0.7k, 1.1k, 1.2k, 1.2k, 1.5k, 3k, 6k, 27k, 33k, 38k, 43k, 51k, 57k, 63k, 64k, 67k, 70k, 92k and 105k across the nineteen runs — median 38k — and in every one of the nineteen it precedes that run's layer-0 wiring median. Sixteen of the seventeen grokkers reach it before their grok; the exception is i0_d2 (105k against a 103.5k grok). The flagship film run is not one of the nineteen and should not be generalised from. An earlier draft put its first full agreement at 60k, six thousand steps after its 54k grok, and used that to explain what D.1's retired frequency-set reading was seeing; direct measurement of its checkpoints does not reproduce it. It is at five-of-six through 33k, six-of-six from 34k to 39k — twenty thousand steps *before* the grok, not six thousand after — and back at five-of-six from 40k straight through its grok. The 60k figure is retired as an erratum and the D.1 explanation that rested on it with it. The five-of-six crossing is much earlier and much noisier (0.4k–65k, median 1.0k against full agreement's 38k) and we do not use it for timing. The *ordering* is insensitive to the choice: at five-of-six as at six-of-six, the sharing rung precedes the run's layer-0 median in 19 of 19 — but so does the pooled-power rung, in 19 of 19, as the table above shows. The 14-of-19 figure is the three-rung statistic including layer 1, not this two-rung comparison, and nothing in the sharing rung improves on the power rung as an ordering. We introduced this statistic to supply a training-side anchor for Section 3.2 and it does not, for two independent reasons. It orders no better than the rung it was meant to replace. And it is not a commitment time: measuring, per run, the fraction of checkpoints *after* the first six-of-six crossing that are still at six-of-six gives a pooled **0.47**, ranging from 0.04 to 1.00 with twelve of the nineteen below half, and thirteen of the nineteen sit at four- or five-of-six at their last saved checkpoint, long after grokking. The property flickers for the whole run rather than latching, so a first crossing is the hitting time of a noisy statistic and not a structural event. **We therefore withdraw the timing column rather than qualifying it**, and with it the reading of this rung as a rung. We flag that this statistic was constructed after the pooled-power rung was retired, so it is post-hoc; we report the ordering, which survives both cuts, and not the timing, which does not — the treatment kill #33 gives its own step-1000 separator.

The full per-run table (every cell computed from the released v2 diaries):

Run	Spectra rung	L0 median	L1 median	Strict ordering
i0_d0	0.7k	73.5k	135k	holds
i0_d1	0.8k	79.7k	141.2k	holds
i0_d2	0.6k	110.3k	133.79k	holds
i0_d3	0.6k	42.7k	147.1k	holds
i0_d4	0.8k	82.8k	138k	holds
i0_d5	0.7k	137.5k	140.1k	holds
i0_d6	0.7k	138k	142k	holds
i0_d7	0.8k	109.7k	142.4k	holds
i0_d8	0.7k	130k	132.5k	holds
i0_d9	0.8k	128.1k	141.8k	holds
i1_d0	0.6k	134.1k	140.4k	holds
i2_d0	61.6k	141k	144.4k	holds
i3_d0	0.6k	134.3k	143.1k	holds
i4_d0	0.8k	25.4k	143.9k	holds
i5_d0	0.8k	145.4k	142.9k	saturated (both medians 138k–147k)
i6_d0	1.3k	146.8k	145k	saturated
i7_d0	0.7k	140.5k	138.2k	saturated
i8_d0	0.6k	145.1k	139.5k	saturated
i9_d0	0.5k	142.9k	142.6k	saturated (censored run)

We label these five censored rather than violating, but the label should not hide the arithmetic: in every one of them the layer-0 median is *strictly greater* than the layer-1 median (145.4k against 142.9k, 146.8k against 145k, 140.5k against 138.2k, 145.1k against 139.5k, 142.9k against 142.6k), so under the unadjusted last-change metric they are inversions of the predicted ordering and not ties. The cohort structure matters too: the ordering holds in 10 of 10 runs of the fixed-initialization family and fails in 5 of 9 of the varying-initialization family, so the three-rung hierarchy is supported primarily within one cohort. All five exceptions have both layer medians inside 138k–147k against the 150k budget — the budget-end saturation, verified cell by cell. Three qualifications belong with that number, and each weakens it.

First, symmetry: i0_d6 (138k/142k) and i2_d0 (141k/144.4k) are *conforming* runs whose medians also sit inside the window. Censoring every in-window run regardless of ordering — the symmetric rule — leaves 12 resolvable runs, and the ordering holds in **12 of 12**. Second, budget pinning: in 11 of the 14 nominal holds the layer-1 median is at or beyond 138k, so for those runs "layer-0 median $\leq$ layer-1 median" is close to automatic once layer 0 resolves early, and the comparison's content is thin. This is the same concession we make for the spectra rung, and it is owed at the top of the ladder too. Third, and most consequential, **the result is cohort-structured**: the ordering holds in 10 of 10 runs of the fixed-initialization family (i0_d0–i0_d9, varying the data seed) and in 4 of 9 of the varying-initialization family, with all five saturations falling in the second family and contiguous within it (i5_d0–i9_d0). We do not know whether that reflects an initialization-seed property of when wiring resolves or a difference in those runs' diaries, and we report it rather than pooling over it. The honest statement of record is therefore: the ordering holds wherever it is resolvable (12/12 symmetric, 14/19 under the registered rule), and its population support is concentrated in one of the two seed families. The table also confirms D.4's competition timing per run: the spectra rung settles at 0.5k–1.3k in 18 of 19 runs, with i2_d0 a lone late-competition outlier (61.6k) whose ordering nonetheless holds. One construction note, owed to the mixed conventions: the spectra rung is a threshold-crossing statistic while the wiring rungs are last-change statistics, and in runs whose competition resolves before the first diary frame the spectra rung is early nearly by definition — the ordering claim's content therefore lives in the wiring rungs' separation and in D.4's suppression-and-recovery dynamics, not in the spectra rung's precise value.

Lock-in is not finality. Tracked against the *final hardened program* rather than argmax stability, layer 0 *touches* 1.000 match-to-final at the 98k checkpoint; one unit (u0_19) then flips a single argument slot away at 99k, leaving the diary's last frames at 0.995, and hardening restores its 98k choice — so even layer 0's finality is a (nearly finished) ratchet. The C.1 excerpt does not include u0_19; its "still churning" annotation sits on the layer-1 unit u1_17, which is the ratchet's specimen rather than layer 0's. Layer 1 accretes from 0.057 at 50k to 0.672 at 100k. Final-matching selections

persist at ∼0.97 per 2k steps while non-final selections churn at 0.46–0.89; there is no discrete event near 90k at all; and the last third of layer-1's final program is supplied by the hardening step itself. A wire can lock, confidently, on a choice that hardening later overrules — which is why the paper's vocabulary separates lock-in (argmax stability) from finality throughout.

D.4 Competition and suppression diaries

The nine diaried runs (i0_d0–d5, i1_d0, i2_d0, i5_d0) show the same three-act population dynamic. The winner-take-most competition among frequency families — a sigmoidal rise in the winners' spectral power — is decided between steps 400 and 1000 in every run under this rise statistic. (D.3's halfway-crossing statistic agrees in 18 of 19 population runs and places one, i2_d0, much later — two statistics on one process; the construction note in D.3 applies.) Transition midpoints (first val-0.5 crossing) span 22k–79k over the same nine runs (test-side: 21.5k–78k, per A.1): the competition and the transition are decoupled by more than an order of magnitude. In **9 of 9** runs, memorization then *suppresses* the winning waves — power dips below the post-competition level with troughs of -0.07 to -0.25 — before recovery through generalization: coexistence, but adversarial. The suppression dip is directly visible on Figure 3a's spectra-share trace, recomputed from raw checkpoints. Which family wins is unpredictable under every initialization predictor we tested, but the *shape* of the winning set is lawful — the winners form aliased-harmonic blocs.

D.5 The parity film

The same filming applied to parity403_i0 across its 115.5k-step plateau, three traces (Figure 3b): the embedding's **relevant-bit share** (squared-power share of embedding columns on the 3 support bits; chance level 3/40 = 0.075), layer-0 **wiring match to its at-grok program**, and the **embedding norm**.

Plateau (1k–80k): relevant-bit share 0.05–0.10 against the 0.075 chance level — at chance to ∼70k (0.070 at 40k, 0.090 at 70k) and drifting to 1.3× chance by 100k — while layer-0 match to the at-grok program sits at 0.05–0.09 over 20k–80k (0.016 at 1k, 0.193 by 90k). The norm rises from 7 to its peak of ∼23 at roughly 40k, mid-plateau, then *erodes under weight decay through the plateau's entire second half* — an Omnigrok-style radial contraction running ∼75k steps before the snap (irregularly: ∼20 at 50k, ∼17–19 through 60–80k, ∼14.5 by 100k) — which is itself relevant to the scale-matching law of B.10. Approach (100k–115k): layer-0 match climbs 0.37 → 1.000 *at the snap*, and 0.281 → 1.000 across the common 20k approach — while the embedding's share drifts only 0.099 → 0.158, from 1.3× to 2.1× chance; the norm reads 14.2 at grok (115.5k). The share is *at* chance through the plateau proper and merely *near* it at the snap, which is weaker than an earlier draft's "at chance at the snap" and is the statement of record (erratum, panel #9). Post-grok: the embedding *organizes* only in cleanup — as the norm collapses a further ∼11×, to 1.28 (∼18-fold from peak to final) — and the donor's relevant-bit share at 150k reaches **0.83** against the 0.075 chance level (squared-power convention; this and the norm trajectory correct two earlier draft numbers — a 0.392 share and a "21 → 1.3" post-grok collapse — that did not reproduce from the raw checkpoints and are retired). **The reference problem, and the matched trace.** Figure 3b sets an endpoint-referenced wiring trace beside a chance-referenced embedding statistic, and that pairing flatters the contrast. Two corrections are owed. First, the wiring's null is not zero: layer 0 selects among 25 signals, so chance match is 0.04, and the 0.05–0.09 band above is *also* near chance — the wiring's informative move is the final ∼25k of the approach, not the whole plateau. Second, the symmetric embedding statistic is not flat. Scoring each checkpoint's embedding against its own at-grok embedding by cosine similarity gives 0.474, 0.538, 0.602, 0.606, 0.884 at 20k–100k on i0 and 0.603, 0.655, 0.724, 0.903 at 20k–80k on i1 — above the wiring's match at every step, and rising throughout. The embedding is converging on its at-grok state during the plateau; what it defers is specifically the *allocation of power to the support bits*, which is what the relevant-bit share measures and what the probe below tests. The inversion is therefore a claim about allocation, not about the embedding being idle, and Figure 3b should be read with the matched trace in mind.

A probe-sensitivity control, and what it costs the claim (kill #30). The relevant-bit share is a *magnitude* measure, so it would read chance even if the support bits were functionally privileged without carrying power. We therefore fit a linear probe from the embedding's output back to each input bit and compared the three support bits against the other 37 — a test invariant to rotations

of the embedding's output space, and so strictly more permissive than column power. The positive control passes decisively: after cleanup the support bits decode at 0.981 and 1.000 on the two trajectories against irrelevant means of 0.871 and 0.821, above the irrelevant maximum (z = +2.5, +3.6), so a null from this probe is interpretable. The plateau result is split, and we report it as a partial falsification of our own registered prediction. On the first trajectory the support bits are indistinguishable from the irrelevant distribution throughout (z = -0.00, -0.06, +0.60 at 40k, 70k, 100k). On the second they are not: by 70k — 25k steps before that run's grok — they sit about two standard deviations above the irrelevant mean (z = +2.01) while the power share reads 0.074 against a 0.075 chance level. The effect is an upper-tail one rather than clean separability: at that checkpoint the best *irrelevant* bit decodes at 0.970 against the weakest *support* bit's 0.961, which is not true after cleanup. The claim that survives is therefore about allocation, not information: parity's embedding does not commit power to the support bits until cleanup, on both trajectories, and on one of the two the information is already linearly present beforehand. Parity's hidden progress, in the sense of what training slowly *builds*, lives in the layer-0 wiring — the inverse of modadd's spectra-first ladder — but "the embedding is doing nothing" is too strong, and we do not claim it.

The seed sweep, and a falsified mechanism of our own. Two further lineages were trained to the full budget under a registration written before launch, predicting that both would grok. Neither did: seed 2 generalizes at $\sim$2k and seed 3 at $\sim$0.5k, both memorizing and generalizing in the same evaluation, so neither has a plateau and neither can test the inversion. Configurations were verified identical apart from the seed, and the support bits are the first three by construction, so only the sampled data and the initialization differ.

Both fast lineages showed their embedding's relevant-bit share leaving chance within 200–300 steps (seed 3 reaching 4.9$\times$ chance by step 300 and 8.9$\times$ by 400) where the plateau lineages sit at chance for 60–117k. We registered that observation as a mechanism — that share at step 500 predicts which regime a run enters — and screened six further seeds at a 10k budget to test it. **It is falsified on both registered clauses.** Seed 6 generalized by 4k with its embedding at 1.1$\times$ chance at step 500, and seed 7 stalled at 1.8$\times$; the predicted thresholds were $\geq 2\times$ and $\leq 1.5\times$ respectively. The group means do differ (3.7$\times$ against 0.98$\times$), and at step 1,000 the two groups happen to separate without overlap — but that threshold was chosen after seeing the data and we report it as a hypothesis for a future registration rather than as a result. What stands is the base rate in Section 3.5 and the plain statement that grok time here is extremely seed-variable.

seed	outcome at 10k	share @500	$\times$chance
4	unresolved	0.044	0.6
5	unresolved	0.070	0.9
6	generalized by 4k	0.085	1.1
7	unresolved	0.133	1.8
8	unresolved	0.042	0.6
9	generalized by 1k	0.145	1.9

A 10k budget cannot distinguish "will grok later" from "will never grok", so the four unresolved seeds are reported as unresolved rather than folded into either group.

Cleanup rewires layer 0 even without grokking. Both fast lineages were trained to the full 150k budget and both migrate away from the program that solved the task. Scoring each against its own *at-solution* checkpoint — B.10's at-grok reference, except that for these two runs the solution arrives with no delayed transition — seed 2 matches at 1.000, then decays to 0.896 by 10k, 0.151 by 20k and 0.036 at 150k; seed 3 decays 1.000 $\rightarrow$ 0.688 by 10k $\rightarrow$ 0.016 by 40k, and stays there. Retention at 150k across four lineages is 49.5% (seed 0), 2.1% (seed 1), 3.6% (seed 2) and 1.6% (seed 3) — three of four keep almost none of it. Because seeds 2 and 3 generalized immediately rather than grokking, post-hoc rewiring of layer 0 is **not** a property of the grokking transition; it happens in runs that never had one. That widens the premise of B.10's at-grok donor law from two lineages to four here — five, once seed 4 is added below — and removes grokking as a necessary condition for it. The law's *consequence* — that a final-checkpoint donor implant finishes below baseline — is still measured on two lineages and is scoped that way.

Seed 1 (replication, registered). A second parity trajectory (parity403_i1, seed 1, identical configuration) groks at 95k (final test 0.997) and replicates every trace: relevant-bit share at chance early and drifting to $\sim$1.9$\times$ chance (0.06–0.14) by the end of the plateau, organizing at and after the snap

(0.183 at 100k → 0.896 at 150k); embedding norm peaking at ∼17.5 mid-plateau (40k), eroding through the plateau's second half to 8.8 at grok, then collapsing to ∼3 — the same peak-erode-collapse timeline as seed 0; and layer-0 wiring converging to its *at-grok* configuration through the approach, 0.06 at 50k → 0.24 at 70k → 0.51 at 80k → 0.93 at 95k, while the embedding is still at chance — hidden progress in the wiring, stronger than seed 0's 0.14 → 0.49. One measurement note: for this seed the at-grok wiring, not the final checkpoint's, is the honest reference — post-grok the wiring migrates *away* from its at-grok program (match 0.83 at 110k → 0.02 at 150k) while accuracy holds and the embedding organizes; cleanup rewires (reported as observed). Match-to-final would score the approach near zero for exactly the lock-in-is-not-finality reason of D.3. Per-seed ladder panels: D.1.

Seed 4 (second replication, registered before filming). A third parity trajectory (parity403_i4, seed 4, identical configuration) memorizes at 2k, sits at chance for ∼116k steps and groks at 118.5k. It is the first plateau-exhibiting lineage found since seed 1, so the film was registered in advance and all four predictions are reported as stated. **The inversion replicates.** Scored on one common window — the 20k of approach ending at each run's grok, against each run's own at-grok program — layer-0 wiring match moves +0.719, +0.589 and +0.609 on seeds 0, 1 and 4, while over the plateau proper (2k to 20k before grok) the embedding's relevant-bit share peaks at 1.42×, 1.12× and 1.73× chance. On all three the wiring's large move and the embedding's are separated by tens of thousands of steps, in that order. One registered clause is missed rather than falsified: we predicted seed 4's share would stay below 1.5× chance through the plateau and it reaches 1.73× at 89k, short of the 2× falsification threshold — it passes 2× only *after* the snap (2.5× at 120k, 5.1× at 134k). The threshold was miscalibrated against our own data, since seed 1 already drifts to ∼1.9× inside its final approach, and we record the miss rather than restating the criterion. The shape of the wiring trace is also not shared: seeds 0 and 1 run flat near zero until the approach, where seed 4 makes an early partial commitment (0.021 at 500 steps → 0.312 by 30k) and then *stalls* in 0.28–0.37 for sixty thousand steps. "The wiring climbs steadily through the plateau" is therefore not what these runs show; what they show is that it climbs steeply across the approach on all three while the embedding does not. The norm prediction failed outright and is kill #35. The fourth prediction holds: by 150k this lineage retains only 0.307 of its at-grok layer-0 program, against 0.495 and 0.021 on seeds 0 and 1, so cleanup rewires layer 0 on all three plateau lineages and the at-grok donor reference of kill #29 is not a property of one run. Its embedding organizes after the snap exactly as the others do (relevant-bit share 0.656 at 150k against the 0.075 chance level) while the norm collapses from 10.3 at grok to 3.21.

Seed 7 (third replication, registered before filming), and the four-lineage summary. A fourth plateau trajectory (parity403_i7) memorizes at 2.5k and groks at 63k — a 60.5k plateau, roughly half of seed 4's, which is why Section 3.5 quotes a range rather than "∼100k". Both registered clauses pass, and this lineage's film is the cleanest of the four. Layer-0 wiring match-to-at-grok climbs steadily across the *whole* plateau rather than snapping late — 0.042 at 500 steps, 0.083 at 10k, 0.177 at 20k, 0.307 at 30k, 0.500 at 40k, 0.729 at 50k, 0.906 at 60k, 1.000 at the grok — while the embedding's relevant-bit share sits in a 1.28–1.56× chance band across the plateau's second half and organizes only *after* generalization (2.4× at 70k, 7.2× at 80k, 12.6× at 150k). Scored on the common window used above:

lineage	grok	plateau	wiring across the 20k approach	embedding band, midpoint to grok−2k	at-grok program retained at 150k
i0	115.5k	111.5k	0.281 → 1.000 (+0.719)	1.12–1.88×	0.495
i1	95.0k	92.5k	0.411 → 1.000 (+0.589)	0.85–2.05×	0.021
i4	118.5k	116.5k	0.391 → 1.000 (+0.609)	1.11–1.88×	0.307
i7	63.0k	60.5k	0.609 → 1.000 (+0.391)	1.28–1.56×	0.016

Cleanup rewires layer 0 on all four, so kill #29's at-grok donor reference is a property of the substrate rather than of one run.

Seed 8, the negative control: what a plateau that never ends looks like. The eighth lineage (parity403_i8) memorizes at 3k and never groks — final test 0.507, chance. It is the first run we have that lets the films be read against the absence of generalization, and we registered two predictions before filming it.

The first passes cleanly and is the control working as intended. **Neither post-grok signature appears where generalization does not.** i8's relevant-bit share stays inside 0.73–1.22× chance from memorization at 3k to the end of the run (whole-run max 1.68×, at step 900), never approaching the 2× line, where every grokking lineage's share ends between 8× and 13× chance. Its embedding norm never collapses either: it peaks at 21.7 near 52k, reads 21.1 at 70k, and finishes at 13.60, against final norms of 1.28, 3.36, 3.21 and 4.17 on the four that grok. The embedding organizing and the norm collapsing are consequences of generalization, and they are absent exactly where it is absent.

The second prediction is the one that mattered, and its answer is negative — see kill #36. We registered that the layer-0 wiring of a run that never leaves the plateau would never settle, measured as churn (the share of the 192 layer-0 argument slots — 48 units × 2 arguments × 2 positions — changing selection between consecutive checkpoints). i8's mean churn over its last 20k steps is 0.0513 per 1k steps against the grokking lineages' mean post-grok churn of 0.0491, so the registered criterion passes — by four percent, and on a baseline we should not have chosen. Post-grok churn is *cleanup rewiring*, which this appendix documents, so it is not a settled reference at all: taken individually, i1 (0.0687) and i4 (0.0601) churn more after grokking than i8 does late in its run. A post-hoc cross-check at a coarser scale, where inter-checkpoint flicker averages out — lag self-match, the share of layer-0 slots holding the same selection at t and t+10k, averaged over checkpoints from memorization to 10k before the grok — separates nothing either: i8's mean of **0.566** on that window sits inside the grokking lineages' **0.596, 0.491, 0.716 and 0.534**. The ordering is insensitive to the window: taking every run from 1k to 140k instead gives i8 0.562 against 0.636, 0.545, 0.711 and 0.665, and i8 still fails to separate. We report the pass as registered and the conclusion as negative, because the criterion, not the substrate, is what passed.

A plateau-side signal that survives, and the three lineages that test it (registered). Every separator this paper has proposed so far has failed to confirm (#33, #34, #36 — B.12 scopes each), and all three were *aggregate* statistics over the whole embedding. Reading the embedding row by row instead finds one that is not. For each of the 24 rows define its support purity `dim_rel[d] = ||W[d,0:3]||^2 / ||W[d,:]||^2` (chance 3/40 = 0.075), and let **P = max_d dim_rel[d]**. A single row becomes a near-dedicated three-bit detector well before the snap, while accuracy is still at chance:

lineage	grok	P >= 0.50 sustained	lead	acc there	P at grok	row
i0	115.5k	95k	20.5k	0.517	0.988	5
i1	95k	78k	17k	0.553	0.781	0
i4	118.5k	87k	31.5k	0.545	0.763	11
i7	63k	50k	13k	0.511	0.928	10
seed 10 (new)	44.5k	7.5k	37k	0.514	0.839	18, then 10

Against the three lineages that never grok inside the 150k budget, P never crosses at all: maxima **0.303** (i8), **0.220** (seed 15) and **0.305** (seed 17), all below the 95th percentile of 0.326 for the max row purity of a random 24x40 Gaussian embedding (20,000 draws; mean 0.222, p95 0.326, p99 0.383). Five of five *plateau* grokkers cross; none of three non-grokkers does. Fisher's exact one-sided test on that separation gives **p = 0.018**; with the single non-grokking lineage the population previously contained, the same perfect separation would give p = 0.20, which is why the two additional lineages (registered and trained under the procedure of A.5 before being scored) carry the result rather than the statistic does.

A sixth grokker, disclosed but not counted. Seed 16 groks at 11.5k by a steady climb from memorization with no plateau, and its P crosses 0.50 at 5.5k — six thousand steps before its grok, in the direction the signal predicts. We disclose it and decline to bank it, because it falls outside both registered clauses: P1 speaks only to lineages that fail to grok, and P2 only to plateau-then-grok lineages. Folding it in would read six of six at Fisher one-sided **p = 0.012**, and we report the registered scoping instead. Seed 16 is also the lineage that sets the 11.5k upper end of Section 3.5's no-plateau range; its 4.5k flat stretch at chance and 9.5k memorize-to-grok delay sit below the ≥20k

plateau threshold fixed in advance in `scratch_parity_seeds.py`, which is why it is binned as no-plateau. That binning is a judgement against a pre-set threshold, and a reader who would draw the line elsewhere should know it is the marginal case.

Two properties of the measure matter. It is computable **prospectively in time** - unlike the match traces above, which are defined against the run's own at-grok program and so cannot be evaluated while a run is in progress. It is **not** blind, and we were not explicit enough about that. P is scored on the known support bits, and its null is the same fixed slice; an observer who did not know which three bits mattered would have to search all C(40,3) = 9,880 triples in each of the 24 rows. That supremum is far larger. On 20,000 random 24x40 Gaussian draws the oracle statistic has mean 0.222 and p95 0.326, reproducing the figures above, while the blind statistic has mean **0.484** and p95 **0.559** - so the 0.50 threshold is crossed by chance in 0.03% of draws under the oracle null and in **31%** under blind search (`scratch_purity_blind_null.py`; the blind per-row maximum is the sum of the three largest squared coordinates, so no enumeration is needed). P is therefore an oracle-conditioned diagnostic, not a feature-discovery detector: it shows that a dedicated row forms before the grok, and does not show that one could be found without knowing what to look for. And it is invisible to both instruments already in this appendix: the dedicated row carries only 2-8% of total embedding power, so a 99%-pure row moves the global share by about 0.08 and vanishes inside the 0.85-2.05x band; and the rotation-invariant probe is blind to it by construction, since a dedicated coordinate is a claim about the basis, and this architecture selects individual embedding dimensions by name.

What "sustained" means, and where it is generous. We record the *first* step at which P crosses 0.50, and that is not the same as holding it. On seed 10 the crossing at 7.5k is held only to 24.5k, decays to 0.399, and is rebuilt on a different row before the grok; i4's 0.42 at 40k is treated as a false start and its 87k crossing recorded instead. Those two are not scored alike, and the seed-10 lead of 37k should be read as an upper bound on the prospective warning it would have given, not as a continuously maintained signal. What the column supports is that a dedicated row appears before the grok in every grokker; the interval over which it persists is not a quantity this table measures.

The claim is one-sided and we state it as such. At a fixed step of 50k the values are 0.22, 0.22, 0.36, 0.52 and 0.20 across i0, i1, i4, i7 and i8 - no separation. Absence at time t predicts nothing; what the population supports is that the signal eventually fires in every lineage that groks and in none that does not. Nor is the rise monotone: seed 10 built a detector on row 18, held it above threshold from 7.5k to 24.5k, lost it to 0.399 as the embedding norm grew from 10.2 to 15.0, and then rebuilt on row 10 before grokking - and i4 shows the same false start (0.42 at 40k, 0.24 by 70k). The honest description is a contested construction: rows attempt the detector, memorization pressure dilutes them, and the run groks off whichever attempt survives longest. *Survives*, not *is read*: the companion registration under kill #40 tested whether layer-0 learns to route through the purified row and found it does not, the read fraction sitting at 0.010–0.021 against a uniform 0.040 and staying at or below uniform at the grok on all four filmed lineages. P is therefore a marker of representational restructuring that correlates with generalization, not a gate the wiring opens. This refines kill #36 rather than overturning it - no point-in-time measurement separates a plateau that will end from one that will not.

The deep film: what four lineages look like side by side. Filming all four plateau lineages together, with layer-**1** wiring and test accuracy on the same axes for the first time, settles one claim and kills three. Method note: we explored on the two lineages already in this paper (i0, i1) and registered two laws before measuring them on the two new ones (i4, i7), so the holdout is real. Both laws failed and are kill #37; the positive result was a contrast specified in advance.

The circuit is largely built before it does anything. At the first checkpoint where test accuracy reaches 0.55 — its departure from chance — layer-0 match to the at-grok program reads **0.39, 0.51, 0.71 and 0.91** across the four lineages. On i7 the wiring passes 0.90, essentially its final program, while accuracy is still 0.542; on i0 it passes 0.90 at accuracy 0.669. This is the paper's "hidden progress" in quantitative form, and we state it as a range rather than a law: we registered a threshold (match $\geq$ 0.40 at accuracy-departure on both holdout lineages) and i4 returned 0.391, missing by 0.009. The registration failed as written and we do not move the threshold or re-specify the window, which is the manoeuvre kill #33 refused.

The within-wiring order is not fixed. We registered that layer 0 would lead layer 1 throughout, as it does on modular addition. It does not: at the first checkpoint where layer-0 match reaches 0.5, i4's layer 1 is **ahead** (0.573 against 0.547). What holds on all four is weaker — by the time layer 0 is essentially complete (≥ 0.9), layer 1 is behind it (0.536, 0.844, 0.865, 0.542). So parity inverts *embedding versus wiring*, and the order **within** the wiring is not a reliable feature of either task.

The one positive result, and it is about the architecture rather than about grokking. Comparing layer-0 with layer-1 churn over the plateau — one contrast, specified before measuring, so no multiple-comparisons problem — layer 1 is the more volatile on **five of five** parity lineages: ratios 2.03, 1.43, 2.14, 1.58, and **1.58 on the lineage that never groks**. Section 4.1's ratchet has been a modular-addition claim on one task family; it now holds on both, and its appearance at the same strength in a run that never generalizes shows it is a property of the substrate rather than a signature of grokking.

What the films do not show. The four traces do **not** collapse onto a universal curve when time is rescaled by plateau length: they are spread across most of the plateau and converge only in its final $\sim$15%. The layer-0 chooser softmax entropy is flat throughout; the released film's `ent0` column, a different quantity, declines on all five lineages (Spearman -0.18 to -0.72) and separates the never-grokker from the rest no better than chance. And a last-mile analysis — which component moves most between layer-0 match 0.9 and the grok, against a matched earlier window — found one tensor moving more on all four lineages, which we do **not** report as a finding: with six tensors and four runs, at least one four-of-four pattern arises by chance about a third of the time, and nothing survives that correction.

E DISTILLATION-PROBE APPENDIX

The printability probe, its press definitions and its instrument limits are documented here; the film itself is Figure 5.

E.1 METHODS DETAIL

The probe, fixed once. For each frozen teacher checkpoint, a discrete student is distilled for 10k steps with the embedding frozen, weight decay off, and KL to teacher logits over all 9,409 inputs; P(t) is the best-of-6-restarts argmax agreement. The schedule was fixed once and never tuned per checkpoint, and the analysis criteria were pre-registered before the frames were scored. The frame grid runs every 5k steps from 25k to 50k and every 10k from 60k to 100k — the grid coarsening after 50k is part of the lag bound's honesty. Instrument lines: cold-restart floor ≈ 0.04 (the same probe applied to an untrained teacher); probe ceiling ≈ 0.72 crisp, 0.775–0.794 entropy-softened. Teacher accuracy per frame: 25k 0.153, 30k 0.203, 35k 0.476, 40k 0.78, 45k 0.898, 50k 0.925, 60k 0.97, 70k 0.975, 80k 0.98, 90k 0.98, 100k 0.98.

The k0/k1 asymmetry. Of the six restarts, one (k0) begins unperturbed at the teacher's own wiring; single unperturbed restarts are far more variable than perturbed ones, at final frames and early ones alike — at the 70k frame, k0 reads 0.483 against k1's 0.830 — and best-of-K is the registered metric. E.2 explains why k0 in particular struggles: starting at the sitter means starting inside a bad hard basin.

The per-τ entropy table. The temperature-artifact deflation is excluded by a paired control: soften the *final* checkpoint to each early frame's logit temperature and press it with the same probe. The claim that survives review is the paired one, and it holds in all six pairs — at every matched τ, the softened final prints above the real checkpoint it is matched to (real early prints: 0.11–0.35 through 45k).

Frame	Matched τ	Softened-final k0	Softened-final k1 (best)
25k	2.2	0.452	0.775
30k	1.8	0.313	0.433
35k	2.0	0.545	0.587
40k	1.6	0.338	0.625
45k	1.3	0.233	0.607
50k	1.8	0.277	0.794

Per-τ paired dumbbell plot and table: fig. A-E1.

Knee and lag, as registered — with the film's corrected numbers. The pre-registered knee criterion specified the first frame with teacher accuracy ≥ 0.90; the 45k frame (accuracy 0.898, best print 0.354) misses the registered threshold by 0.002 and is reported as the frame that just misses; at 40k the pair reads 0.78 against 0.29, and no frame through 45k prints above 0.36. Warm prints then clear 0.7 at the 50k knee itself (0.725) — in the same frames as layer-0's commitment — while *cold restarts* flip only between the 50k and 60k frames ($0.19 \rightarrow 0.716$). The measured lag is therefore bounded by one frame interval (5k) at this grid. (These values are read against the released film record and correct an earlier draft's 45k best print of 0.338 and its $\geq$10k lag bound, which had anchored the print's arrival on the cold-restart flip rather than the warm print at the knee.)

The late segment. Two presses appear in this appendix, and "crisp" always means the teacher's *unsoftened* logits. The **film press** is the registered instrument above (10k steps, best-of-6 restarts); its crisp best-of-K at the 100k frame, 0.72, is the ceiling line in Figure 5, and its individual 80k restarts reach 0.800. The **seed-matched A/B press** is a separate 3k-step, single-restart probe built to isolate the softening variable, and all numbers in this paragraph are its readings. Past 80k the limiting factor is the probe: under the A/B press the crisp 100k checkpoint prints at only 0.514 though the teacher's function is essentially finished by 80k (98.86% agreement between the 80k and 100k functions; the 80k print scores 0.8645 against the 100k teacher). The A/B closes the case: re-softening the 100k teacher restores 0.698, against 0.734 for the crisp 80k frame under the same press (and the entropy table's 50k k1 of 0.794 sits beside the film's crisp-80k k1 of 0.800). The mechanism is dark-knowledge starvation: the runner-up soft mass the KL probe runs on collapses $5.7\times$ between 80k and 100k ($0.0076 \rightarrow 0.0013$), and the straight-through tail breaks on ultra-sharp targets — terminal KL spiking $6.6 \rightarrow 7.66$ where the softened probe descends smoothly, $5.0 \rightarrow 1.59$. All late-segment claims anchor at $\leq$ 80k.

The rescued-network row. The Section 3.1 rescued network, pressed with the same recipe: hard 0.8816 against soft 0.8829 — faithful, at $\approx$88%, well below the 98% prints of the long-consolidated networks; printability tracks consolidation age (cross-referenced from E.1 and B.7).

E.2 Five-drafts autopsy and multiplicity widening

At the 40k frame the probe returns five distinct printed programs — five layer-1 programs on one shared layer-0 — reported as **measured multiplicity and nothing more**: a stronger reading (superposition of drafts in the teacher) would need a synthetic-mixture control not run. Pairwise agreement among the five drafts is 0.14–0.30 at 40k; against the 50k print, 0.17–0.27; cross-restart layer-0 agreement is already 0.73 at 25k while layer-1 runs $0.43 \rightarrow 0.64$ — the student's layer-1 margin is the printability bottleneck. Prints track the teacher's *current* layer-0 faithfully, at 0.71–0.92 per frame: what looked like non-convergence toward the final program was the sitter's own layer-1 ratchet still accreting (D.3), not print failure. Through consolidation the multiplicity *widens*: warm-print layer-1 agreement falls $0.607 \rightarrow 0.398$ while the shared layer-0 rises to 0.97 — the drafts diverge from one another even as each closes on the final function. Over the same interval the teacher's clock-decomposability falls $0.512 \rightarrow 0.155$ (mass moving into interaction structure) — flagged speculative. The transcription specimen closes the subsection: restart 100k_k2 achieves a near-perfect copy of the teacher's wiring at fidelity 0.138 — a print can be verbatim and worthless, the anti-copy result in specimen form (fig. A-E2).

E.3 Clock-coordinate error analysis

Early print errors are tight phase blur on the f24 clock, not output-space noise. In output space, early-print errors look unstructured (anti-near); re-coordinatized as f24 clock ticks, mean —tick error— runs 6–9.6 against a uniform baseline of 24.2, shrinking monotonically to 2–5 by 60k — early prints carry the f24 wave at 60% amplitude with blurred tick registration, and "blurred truth" holds in clock coordinates only. Residual print errors are wave-structured at the *minority* harmonics: f15/f27/f35 carry 0.6 of the error-profile power, with error bands at the f15 modes 15, 30, 45 — prints fail where the f15 family works (C.3) (fig. A-E3).

E.4 FREQUENCY FATES

Two fates, from opposite ends of the head's spectrum. **f15 takeover:** the probe loses f24 preferentially, and by the 90k frame warm prints are f15-dominant (0.81–0.98 f15-dominant against the teacher's f24) — the drafts sit on an f24:f15 dial, some amplifying the minority wave above the teacher's mix (the dial reading is flagged speculative). **The f35 seam:** f35's *wires* print — the e1 selections are copied — but its *function* never does, ≤ 0.03 through 100k: the ablation-invisible third family (C.3) is also print-invisible, two independent instruments unable to see f35's contribution, closing C.3's vestigial-family verdict from the probe side (fig. A-E4).

E.5 THE MORAL, STATED PLAINLY

The probe reads the teacher's doubts, not its answers; by 100k there are none left. That sentence is the entire late-segment story of E.1, and it is why P(t) is reported raw against instrument lines rather than normalized: the ceiling is itself checkpoint-dependent, and a normalized score would hide the instrument limit it is our job to report.

Two postmortems complete the record. **The anchored film.** An earlier film design chained each frame's press from the previous frame's student; its arbiter control showed the chained frames matching the inherited 25k crystal at 68–94% against only 54–62% to their own checkpoints — the continuity was manufactured by the anchor, and the anchored-film method was retired before any of the printability claims were scored. Every number in this appendix comes from the fixed independent-press design of E.1. **The cold-restart capture trap.** The cold-restart curve dips $0.716 \rightarrow 0.282$ at 90k and recovers to 0.643 at 100k via an idiosyncratic pure-f24 clock that ignored the teacher's wiring — half-captured restarts fail where the fully-original one succeeds. This is one restart per step (n = 1, speculative, replication script staged and unrun — labeled exactly so).

F INIT-ENCODING DETAILS

The coalition census, the permutation null behind it and the trajectory statistics that accompany it are documented here.

F.1 THE HE-PREDICTION TEST AT SCALE

[He26] prove, for decoupled two-layer gradient flow at small initialization, that per-neuron winning frequencies are determined at initialization by spectral magnitude and phase alignment. We tested the prediction at scale on 1,737 committed neurons across 32 standard MLPs, with the chance rate stated: init-dominant equals final-dominant **8.8%** of the time against a $1/48 \approx$ **2.1%** chance rate — 4.2× chance; the final winner sits in the init top-4 **25%** of the time (3× chance); but the final winner's **median init rank is 12 of 48**. Real bias, not determinism, in the standard regime.

F.2 THE INIT-ENCODING AXIS (S-A)

The axis runs from [Barak22]'s parities — where the solution is encoded in the initial population gradient — through [He26]'s decoupled neurons, provably init-determined, to our coupled coalitions, where no initialization predictor works at all. The reading: **coupling destroys init-determinism.** Our predictors-negative result (F.3) is then not a contradiction of [He26] but a demonstration that their decoupling assumption is load-bearing — in a substrate whose competition dynamics are measurably coupled (F.5's coalitions), winners are made by symmetry-breaking, not born, while in their decoupled setting birth is destiny and in the standard MLP regime (F.1) it is a 4.2× bias.

F.3 THE THREE INITIALIZATION PREDICTORS, ALL NEGATIVE

Three pre-registered initialization predictors, each a candidate for forecasting which frequency families win the competition, each tested across all three fully-diaried seeds: (i) a frequency's **initial spectral power**; (ii) its **initial cross-block phase coherence**; (iii) its **alignment with the first gradient** (first-gradient spectral power). All three fail to predict the winners — 3/3 negative. The

negatives are not artifacts of init drift: reconstruction of each run's initialization is bit-exact against the saved step-0 checkpoints (6/6 runs, maximum difference 0.0).

F.4 The init-projection test

In the style of [Liu22]'s Appendix K, we project the final winning frequencies onto the initialization: the winners' mean init-projection is 0.0846 against a 0.0834 baseline over non-winners (uniform 4/48 = 0.0833), p = 0.29 — null. The init-projection lottery reading does not transfer to this substrate.

F.5 The coalition census, with its null model

Winning frequency sets in this substrate are base-plus-aliased-harmonic families: **35 observed harmonic links against 11.5 expected**, $p < 5 \times 10^{-5}$, in 9 of 9 diaried runs. The two expectations are not on a common per-run scale, and we flag it rather than smooth it over: 11.5 across 9 EML runs is 1.28 links per run where 14.3 across 28 MLPs is 0.51, so the raw counts are not directly comparable across substrates and the substrate claim rests on each census against its own null ($p < 5 \times 10^{-5}$ and p = 0.69) rather than on 35 against 13. Why the per-run expectations differ is a question for the released census code; we have not reconstructed it. The null model is stated: expected link counts come from a permutation test over random frequency 4-subsets, which supplies the 11.5 (EML) and 14.3 (MLP) expectations. The dynamics agree with the census: linked pairs rise together (mean trajectory correlation +0.264) while unlinked pairs compete (-0.173) — a difference of +0.437, p = 0.0003 (n = 40 linked vs. 95 unlinked pairs). The same census on 28 MLPs trained on the same task finds nothing: 13 links against 14.3 expected, p = 0.69. The coalition structure is a property of this substrate's competition dynamics, not of the task — which is why the main text scopes it accordingly and uses it only where substrate-internal dynamics are at issue (the coupling premise of F.2). Which family wins remains unpredictable (F.3); the *shape* of the winning set is lawful.

G Extended related work

The neighbours below are placed here rather than in Section 5 because each is a one-clause relationship rather than a competing account; the five lines Section 5 does carry are the ones our controls are designed against.

Interventions that act before or during training. (Minegishi et al., 2025) apply a training-discovered mask at the original initialization and retrain; (Xu et al., 2025) initialize a target's embedding from a weaker model's; (Howe, 2026a) and its companion (Howe, 2026b), one author rather than independent lines, shape representations *during* training under norm-matched controls, finding that structure rather than norm controls the delay and that the prior is needed only in an early window — the training-time cousins of our class result (Section 3.2) and our phase-dependence (Section 3.3). All act at or before the plateau. With our controls added: a mature representation given at birth prevents the plateau, given mid-plateau ends it, and nothing else does.

Component-resolved interventions elsewhere. (Gomezjurado Gonzalez, 2026) is the closest independent relative of our battery and the reason Section 5 no longer claims that no prior intervention dissociates components. On one-step Collatz prediction in an encoder-decoder transformer it fixes a converged encoder and trains a fresh decoder from random initialization, reaching 70% accuracy at step 24,000 against 66,000 for joint training from scratch (the 2.75x), while a transplanted decoder actively harms and freezing the encoder beats joint training 97.6% to 86.1%.

Four differences keep the two instruments apart, and we state them because the overlap is real and we would rather bound it than argue about it. Their donor comes from a separately trained model where ours is the host run's own future, so the cross-model correspondence

question their design must absorb does not arise in ours. Their recipient is randomly initialized where ours is a mid-plateau checkpoint that has already memorized its training set - a fresh network has no wrong solution to dislodge, which is the whole content of the step-100 dip every one of our implant arms shows (B.8). They freeze and train the complement where we fine-tune everything, so their statistic is final accuracy under a frozen prior and ours is rebuild time under a matched optimizer. And they compare two monolithic components with no parameter-count control, where our five arms include a readout head carrying exactly the embedding's 4,656 weights.

What survives all four is the finding itself, and it points the other way from ours: their delay "reflects limited access to already learned structure rather than failure to acquire that structure", the encoder developing parity and residue structure early while output accuracy stays near chance. Ours is a plateau spent building the input representation. We read the disagreement as support for the task-dependence thesis of Section 6 rather than as a contradiction - two task-architecture pairs, the same question, two answers - and as the third such pair after modular addition and sparse parity.

Transferability of a finished circuit. (He et al., 2026b) asks whether grokked transformers are worth the wait on compositional two-hop reasoning, and reports that "even a mature circuit exhibits limited transferability when integrating new knowledge". That is a cousin of our at-grok donor law from a different direction: they find a finished circuit transfers poorly into a new integration context, we find a finished circuit transfers poorly into its own lineage's past once cleanup has rewired it. Neither result is about the other's setting, but together they caution against treating a converged circuit as a portable object.

At-grok measurement validity. (Truong, 2026b) is a companion to (Truong, 2026a) by the same author rather than an independent line, and audits the measurement practice both that paper and ours rely on. It reports that representation metrics keep moving for at least 10k steps after the grokking transition, with effective rank read at grok overstating the final compressed value by 3-5x in MLPs and 1.3-1.5x in transformers, and that reading a metric at grok "can reverse the dependence one set out to characterize." This is the descriptive antecedent of two of our findings: the at-grok donor law of B.10, where a donor circuit taken at the final checkpoint no longer solves the task and finishes below baseline, and D.5's observation that parity's embedding allocates power to the support bits only during cleanup. All three say the same thing from different directions - a checkpoint taken at grok is not a converged one, and which checkpoint you read is a methodological choice with consequences.

Efficiency, exactness and stitching. (Varma et al., 2023)'s efficiency competition names the selection pressure that our transplant names the component of. (Nanda et al., 2023), (Gromov, 2023),

(Zhong et al., 2023) and (Michaud et al., 2024) recover algorithms by reconstruction from outside; ours is a

native print, which is what licenses Appendix C's joint-ablation anatomy. (Truong, 2026a) and our transplant are both trajectory-level relatives of model stitching (Lenc & Vedaldi, 2015; Bansal et al., 2021), which splices across networks where we splice within one. (Sivasankar, 2026)'s Fourier synchronization metric leads generalization by 500-3,000 steps, answering (Nanda et al., 2023)'s ex ante timing problem at population level where Section 4's diaries answer at wiring resolution.

Representation-timing accounts. (Kumar et al., 2024)'s lazy-to-rich transition and (Thilak et al., 2022)'s slingshot effect describe the same relocation of learning that our ladder times at wiring resolution. Both are observational where ours is surgical: they identify when representation change happens, the transplant asks what a host cannot proceed without.

Reconstruction versus native print. (Nanda et al., 2023) recovers modular-addition algorithms by fitting them from outside, reaching 93-98% fraction-of-variance explained; (Gromov, 2023), (Zhong et al., 2023) and (Michaud et al., 2024) are further reconstructions in the same idiom. The distinction that matters for Appendix C is that our print is not a reconstruction at all - the deployed object *is* the discrete program, so joint ablation of a unit family is an experiment on the network rather than on a model of it. The price is the 98.95% translation epsilon, which Section 6 lists as a limitation and every anatomical claim inherits.

Idealized reconstructions from memorizing models. (Swaroop, 2026) builds an idealized network from a *memorizing* model's Fourier parameters and reaches 95.5% test accuracy. This cuts both ways and we report it as such. It is the observational analogue of our first ladder rung — the spectral structure is already present and already sufficient long before the network uses it — and our transplant supplies the causal version, since a memorizing host handed that structure ends its plateau rather than merely being shown to contain it. But it is also the sharpest external argument against reading the plateau as the *acquisition* of that structure: if a memorizing model's parameters already support 95.5%, what the plateau is spent on cannot simply be building them. Our own D.1 (winning-family power share 0.73 at step 1k) points the same way, as does D.3's finding that cross-position sharing flickers throughout training rather than being acquired and held across the plateau, which is why Section 3.2's claim is stated as sufficiency.

Hardening and acceleration techniques. (Yousefi et al., 2025) is the nearest relative of our hardening recipe, deployed to close a train-test efficiency gap where ours is a fidelity instrument whose only job is to make a network printable without changing what it computes (Section 2 controls both fidelity confounds). (Lee et al., 2024) accelerates grokking by filtering gradients over *iteration* frequency; we decompose over *input* frequency. The two senses of

"frequency" are unrelated and the coincidence of vocabulary is worth flagging.

Initialization and parity. (Barak et al., 2022) established the sparse-parity setting and the hidden-progress framing our parity films operate in, and (He et al., 2026a) supplies the init-encoding prediction that F.1 tests at scale (and largely fail to confirm in the standard regime).