SAME ANSWER, DIFFERENT FUTURES: JUDGING LATENT STATES BY THEIR CONTINUATIONS IN LOOPED TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural network states are often judged by their current answer: a state moved into another input earns credit if the output stays correct, and computation stops once outputs settle. For a latent state updated step by step, a correct answer need not last. We study looped transformers trained on finite-group composition, a state-tracking task with provably answer-preserving edits, so states can be transplanted across edits. In every model that passes our accuracy gate, transplants decoding the correct answer turn wrong within several updates far more often than untouched runs, then almost always recover. In models whose errors come later for edits farther from the answer, the transplant's difference reaches the readout mainly through intermediate positions, and delaying it midway delays most errors about as much. This timing gives an edit-aware recipe for reusing latent states: reuse the unedited prefix exactly, warm-start the rest, and count stable outputs toward an exit only after the edit's difference reaches the readout. In those models, the recipe uses less computation than recomputing and, unlike prefix caching alone, exits sooner on average, without more wrong stops. Latent states in these looped transformers should be judged by their continuations.

1 INTRODUCTION

Spending more computation at inference is now a standard way to improve reasoning, and a growing family of models spends it by updating an internal state step after step rather than by writing out intermediate text (Geiping et al., 2025; Prairie et al., 2026). Practitioners act on such states before the computation ends: they patch them into another input to locate what they carry (Heimersheim & Nanda, 2024; Zhang & Nanda, 2024), reuse them to save work (Geiping et al., 2025), and stop once the output looks settled (Schuster et al., 2022; Geiping et al., 2025). Each use reads a correct answer now as a promise about later ones; for a state updated again, the time of reading is part of the claim.

We study looped transformers, which apply one block repeatedly to a latent state and can be read after every update (Dehghani et al., 2019; Geiping et al., 2025; Prairie et al., 2026). We train them on finite-group composition, which formalizes state tracking (Liu et al., 2023; Merrill et al., 2024) and whose algebra supplies edits that provably keep the answer. We move a latent state mid-computation from an edited problem (the *donor*) into the original one (the *recipient*), a *transplant*, and follow its readout against the recipient's *untouched run* (Figure 1).

Correct now, often wrong later. In every checkpoint that passes our accuracy gate, answer-preserving transplants that decode the correct answer at insertion turn wrong within several updates far more often than the untouched run, then almost always recover (Section 3.1). These errors are transient: checks at insertion and at the end both miss them. Patching results are known to depend on the replacement (Zhang & Nanda, 2024); here they depend on the comparator down to the sign: a fixed mean of other problems' states settles on the correct answer sooner than donors whose edit changes the answer but later than donors whose edit keeps it (Section 3.2).

When a correct state goes wrong. By design, a donor differs from its recipient only in the operands it edits, so the transplant's difference starts there. In checkpoints whose errors come later for edits farther from the answer (*edit-ordered*), the readout barely reads this difference at insertion: although

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.

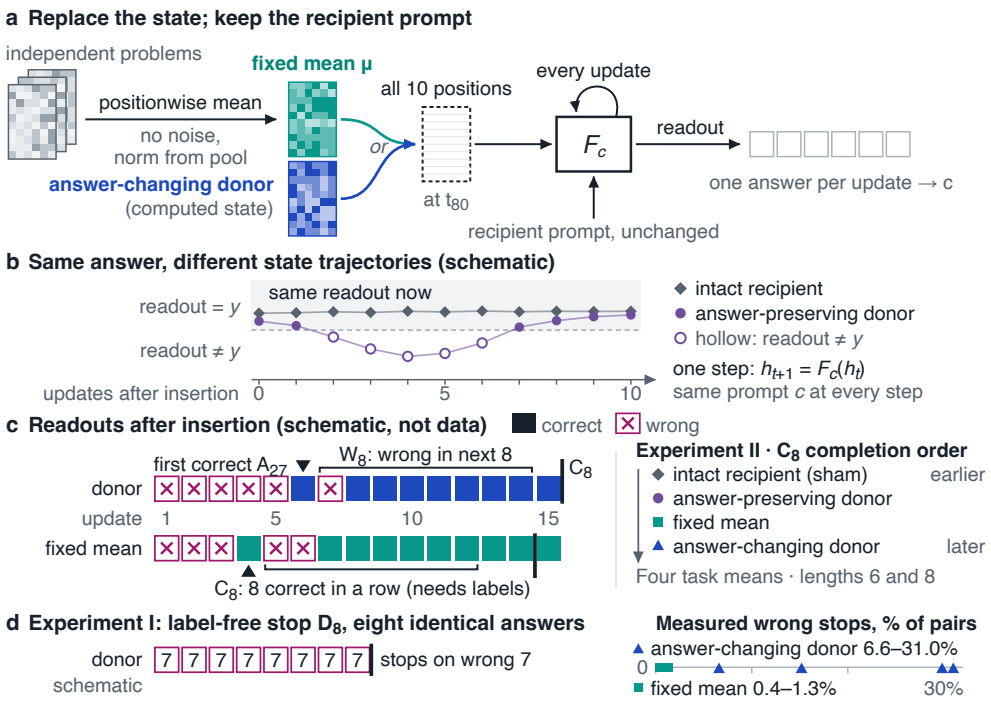


Figure 1: **Same answer, different futures.** A transplanted state that decodes the correct answer at insertion can turn wrong several updates later and then recover, while the untouched run stays correct; reading the present answer cannot tell them apart. (a) t_{80}, insertion time ($\geq$80% accuracy); answer-preserving donors (drawn in b) enter the same way. (b) Schematic; intact recipient, the untouched run. (c) A_{27}, first correct update, up to 27; W_8, wrong readouts in the next eight; C_8, settling time; the ladder gives only the order, the same for every task mean at both lengths. (d) The eight-repeat exit (D_8) can stop on a wrong answer. Experiment I compared answer-changing donors with mean-based starts (d, length 8); Experiment II added answer-preserving donors on fresh inputs (c; its wrong stops, a separate range: Section 3.3).

the answer position can attend to the edited operands from the first update, the difference passes mainly through intermediate positions to the positions the readout uses (Figure 4). Errors arrive with it, and delaying it midway by a few updates delays most of them by about as many, an intervention separate from how these checkpoints were selected (Section 4). Transient chaos, reported in looped models (Lai et al., 2026), would also produce late errors, but its fractal boundaries gain switches when refined; our error boundaries gain none at the resolution we test (Section 5).

An edit-aware recipe for reusing latent states. Recurrent-depth language models can warm-start each token's recurrence from the previous token's final state and exit once successive outputs change little (Geiping et al., 2025); warm-starting after a prompt edit is a transplant from the old prompt into the new one. The pre-edit answer can then linger: in our models, an exit on repeated outputs stops answer-changing transplants on wrong answers more often than answer-preserving ones or a fixed mean (Section 3.3). We turn the difference's arrival time into a recipe: reuse the states left of the edit exactly, as causal attention allows (prefix caching at every update); warm-start the rest; and (the new step) count stable outputs toward an exit only after the edit's difference reaches the readout. In each edit-ordered checkpoint, the recipe uses less computation than recomputing, stops wrongly no more often and, unlike prefix caching alone, exits sooner on average (Section 6). A *continuation check*, comparing a state's next updates with the untouched run's, audits patched states offline.

1. **We show** that correct transplants in our models often err, then recover, and that whether a replacement looks helpful depends on its comparator (Section 3; Figures 2 and 3).
2. **We trace** the difference in the edit-ordered checkpoints from the edited operands mainly through intermediate positions to the readout, and find that refining the error boundary adds no switches at the resolution we test (Sections 4 and 5; Figures 4 and 5, Table 1).
3. **We build** an edit-aware recipe for reusing latent states and a continuation check for replaced states, and evaluate both on saved runs (Section 6; Figure 7, Table 2).

2 SETUP: LOOPED TRANSFORMERS, TRANSPLANTS AND MEASURES

Under prompt c (inputs e_c), each *update* rewrites every position, $h_{t+1} = F(h_t, e_c)$; the readout runs the exit blocks over all positions and reads the last. Our looped transformers (the Parcae looped language model architecture (Prairie et al., 2026), trained on random update counts; Appendix H) compute ordered products $P = a_1 \cdots a_L$ in $\mathbb{Z}_{61}$, $(\mathbb{Z}_2)^6$ (XOR), D_{31} or A_5 for $L = 8$ or 6. We analyse the 16 of 20 trained checkpoints (seeds s0–s4 per task) that pass an *accuracy gate* (Appendix E). Positions $0, \ldots, L+1$ hold the start token, the operands and the final '=', the *answer position*. A transplant swaps in the donor's state at the *insertion time* (by default the first of three consecutive updates at $\geq 80\%$ accuracy) and continues under the recipient's prompt; the *sham* is the untouched run, and the *lag* counts updates since insertion. A pair is *both-correct* if the transplant and the untouched run decode the correct answer at insertion; an *error* is a later wrong readout. The *settling time* is the lag ending the first run of eight correct readouts; the label-free *eight-repeat exit* stops where the first run of eight identical ones ends, so it can stop on a wrong answer (C_8, D_8 in figures). Attention is causal, so no other position reads the answer position, and positions left of the first edited operand match the untouched run up to round-off for any edit.

Donors, comparators and checkpoint sets. In Experiment II (both donor types, fresh inputs), donors edit the first two operands, the *edited operands* (the *seam*), and keep or change the answer (k, g nonidentity): $c^+ = (a_1 k,\, k^{-1} a_2,\, a_3, \ldots, a_L) \mapsto P$ and $c^- = (g a_1 k,\, k^{-1} a_2,\, a_3, \ldots, a_L) \mapsto gP \neq P$ (1), giving *answer-preserving* and *answer-changing* transplants. The *fixed mean* starts from the positionwise mean of both donor types' states on 64 other problems per block, at their median norm (Appendix B). Mechanism analyses centre on the 9 *edit-ordered checkpoints* (the *roster* in figures), whose errors came later for edits farther from the answer in a sweep run before the clamp, patch and delay experiments; 4 meet every edit-order criterion we tried with room to spare, a sensitivity set fixed after the first mechanism runs (Appendix G.3).

Evidence levels. *Frozen:* Experiments I (answer-changing donors vs. mean-based starts) and II froze inputs, code, checkpoints and contrasts before measurement (not externally registered). *Descriptive:* post hoc analyses of those runs, including all sham values. *Exploratory:* later runs (Sections 4 and 5; Section 3.1's update-40 control) and Section 6's post hoc re-analyses; script predictions fixed before running but after earlier runs (not blind) are scored in Appendix G.3.

3 A CORRECT ANSWER NOW CAN TURN WRONG LATER

3.1 TRANSPLANTS THAT START CORRECT ERR, THEN RECOVER

Numbers are Experiment II at length 8, pooled over pairs unless given as ranges. Among the 6,582 both-correct pairs, 53.36% of answer-preserving transplants err within eight updates, against 0.84% of untouched runs (length 6: 56.29% vs. 0.90%; Figure 2a). The gap holds in each of the 16 checkpoints at both lengths (12.8–90.7% vs. at most 3.4% at length 8; Figure 2b). It holds for confident starts too: in each within-checkpoint quartile of the correct answer's logit margin at insertion, 49.1–56.1% of transplants err vs. 0.2–2.3% of untouched runs, and 47.0% err even when Lemma 1 (Section 4) certifies the correct readout at insertion. It also holds after the runs settle: in a later run, transplants inserted at update 40 err about as often as early ones (58.0% vs. 59.1% of 828 pairs, lags 1–20; ratio 0.982, 90% interval [0.968, 0.996]). Errors are transient, beginning a median 2–7 updates after insertion (by task), and 99.6% of erring transplants read the correct answer again at lag 35 (99.8% at length 6; $\geq$96.6% and $\geq$98.3% per checkpoint).

3.2 WHETHER A REPLACEMENT HELPS DEPENDS ON THE COMPARATOR

The same fixed mean wins or loses according to the donor type. In frozen contrasts at length 8, the fixed mean settles 0.67–0.82 updates sooner than an answer-changing donor but 2.86–5.57 updates later than an answer-preserving donor that edits the same two operands (Figure 3a,b). Their simultaneous 95% intervals exclude zero in every task and at both lengths, with opposite signs; one frozen criterion, that the mean would also beat answer-preserving donors, is not met (Appendix B). Experiment I (answer-changing donors; Figure 3e,f) agrees: a mean plus noise beat the donor in all four tasks, and, post hoc, a noise-free mean settled sooner but erred more after its first correct readout, as Experiment II's frozen replication confirmed in every task and length.

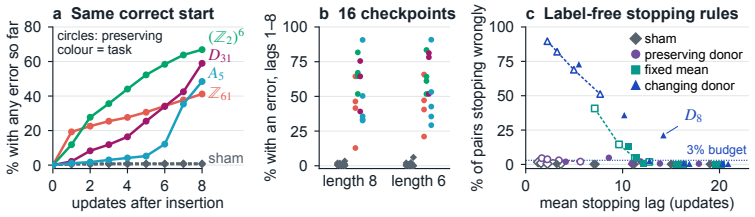


Figure 2: **Transplants that start correct err far more often than untouched runs.** Experiment II, length 8 except (b). (a) Both-correct pairs: share erred so far, answer-preserving transplants (circles) vs. untouched runs (sham); colour, task. (b) Per checkpoint, share erring by lag 8. (c) Label-free exits, four-task means, all pairs: filled, k repeats after b updates (D_8: k=8, b=0); hollow, $\mathrm{KL}(p_t \| p_{t-1}) < \tau$; dotted, post-hoc 3% wrong-stop budget.

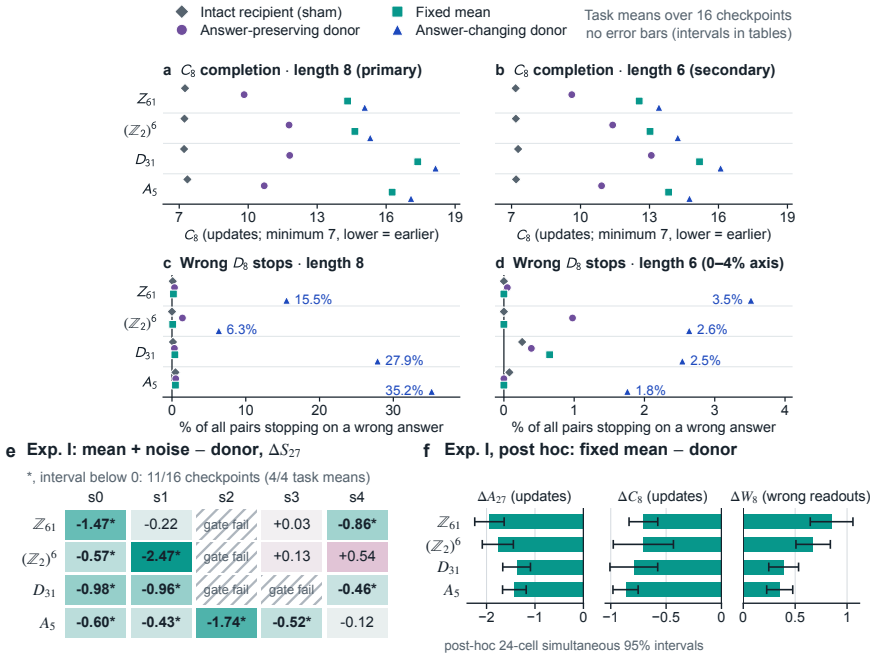


Figure 3: **Whether a replacement helps depends on the donor type; a stable output can be wrong.** Experiment II task means (intervals: Appendix B): (a,b) settling time C_8; (c,d) wrong stops of the eight-repeat exit D_8. (e,f) Experiment I, answer-changing donors: (e) mean plus noise minus donor, per training seed s0–s4; S_{27}, start of the last correct run up to lag 27; hatched, gate failed; (f) fixed mean minus donor, post hoc; A_{27}, first correct lag; W_8, wrong readouts in the next eight.

3.3 A STABLE OUTPUT CAN REPEAT THE DONOR'S ANSWER

Answer-changing transplants stop on wrong answers. The eight-repeat exit stops 6.35–35.20% of answer-changing transplants on a wrong answer at length 8 and 1.76–3.52% at length 6, against 0.33–1.42% for answer-preserving donors and 0.10–0.47% for fixed means (length 8; Figure 3c,d); in both frozen contrasts, the answer-changing donor stops wrongly more often in every task and at both lengths. Per task at length 8, 55–98% of these wrong stops repeat the donor's answer, which lingers, stable, at the readout. A KL exit at the threshold 5×10^{-4} that Geiping et al. (2025) used for their language model, applied offline to our models without burn-in, stops 64.1% of answer-changing pairs wrongly (four-task mean), and thresholds from 10^{-4} to 10^{-1} stop 52–90% (pooled over pairs; four-task means in Figure 2c). The transient errors of Section 3.1 rarely stop the eight-repeat exit: among both-correct pairs, it stops answer-preserving transplants wrongly in only 0.05–1.43% per task, so they matter mainly for patching.

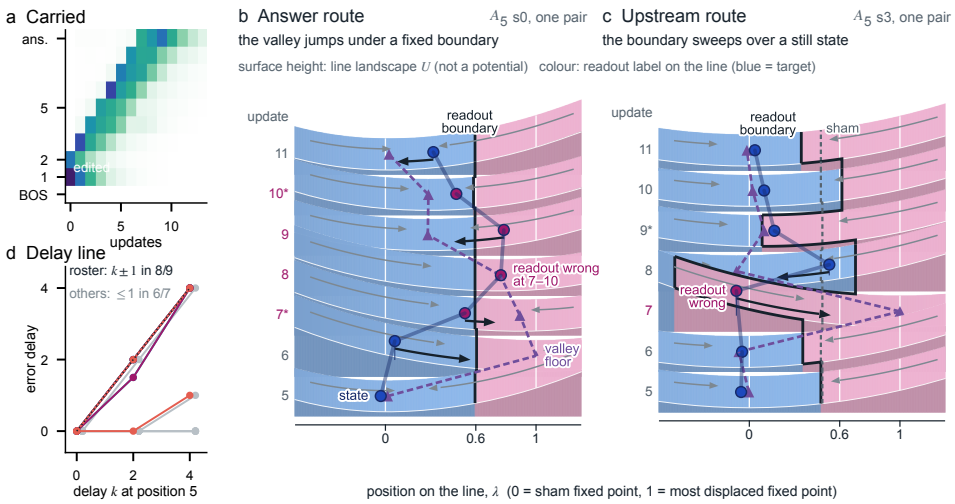


Figure 4: **The difference reaches the readout late, and errors arrive with it.** Length 8, operand-only transplants; A_5 training seeds s0 (a, b) and s3 (c), chosen by us, each with its first both-correct pair that errs (fixed rule). **(a)** Difference from the untouched run (sham) per position (BOS: start token; ans.: answer position). **(b, c)** The answer position (other positions held) on one line of its states, one slab per update, from the untouched run's fixed point at insertion (λ=0) to the transplant's most displaced one (λ=1). Height: measured one-update steps, projected and integrated (downhill by construction). Black: readout boundary; navy: the state, projected (magenta: readout wrong; $*$: unlike the colour beneath). **(d)** Delaying position 5's difference by k updates: median per-pair shift of the first error, edit-ordered checkpoints (roster; colour: task) and others (grey).

4 The difference reaches the readout via intermediate positions

Causal attention makes $F_c = F(\cdot, e_c)$ a skew product in $h_t = (u_t, z_t)$ (upstream positions $0..L$; answer position): u_{t+1} depends on u_t alone and $z_{t+1} = f_c(u_t, z_t)$. The rule is fixed; the effective map $f_c(u_t, \cdot)$ and readout boundary move with u_t. States keeping target y for H more updates form $\mathcal{R}_y^H = \bigcap_{0 \le t \le H} F_c^{-t}(\mathcal{R}_y)$, with $\mathcal{R}_y$ those reading it now; a both-correct transplant erring by lag H starts in $\mathcal{R}_y \setminus \mathcal{R}_y^H$. Proposition 1 (Appendix F.1) splits the transplant's logit change exactly into an *answer route* stored in z_t and an *upstream route* read from u_t: an accounting identity, not a forecast.

Two facts explain when late errors appear. The readout errs only once the logit change the difference causes exhausts the untouched run's margin (Lemma 1); and in the 9 edit-ordered checkpoints (our scope; length 8 unless stated), the difference reaches the positions the readout uses (which differ by checkpoint) mainly through intermediate positions $3..L$ (Figure 4a).

4.1 It starts at the edited operands and is barely read at insertion

By design, a donor differs from its recipient only in the first two operands (Equation 1), so the transplant's difference is concentrated there (0.57–0.97 of its squared size at insertion; medians, all 16 checkpoints). **At insertion the readout barely reads it.** Let $m_t > 0$ be the untouched run's lead of target y over the runner-up logit at lag t, m_t^v the transplant's, d_t the transplant's logits minus the untouched run's, and $c_t = (m_t - m_t^v)/m_t$ the margin consumed. A transplant of the edited operands alone (*operand-only*; "seam only" in figures) has median $|c_0|$ 0.003–0.026 in the edit-ordered checkpoints, and the lemma below certifies its readout at insertion in 0.81–1.00 of pairs.

Lemma 1. *The transplant reads a wrong label at lag t only if $c_t \ge 1$, and $c_t \le \mathrm{osc}(d_t)/m_t$ with $\mathrm{osc}(d) = \max_j d_j - \min_j d_j$; so $\mathrm{osc}(d_t) < m_t$ certifies a correct readout (Appendix F).*

The edited operands suffice. Both-correct operand-only transplants err at lags 1–20 about as often as full ones or more (full minus operand-only rate -0.02 to $+0.08$, A_5 s0 -0.45); the answer position's difference alone errs about as often as the untouched run (excess $\le +0.04$; Figure 5a).

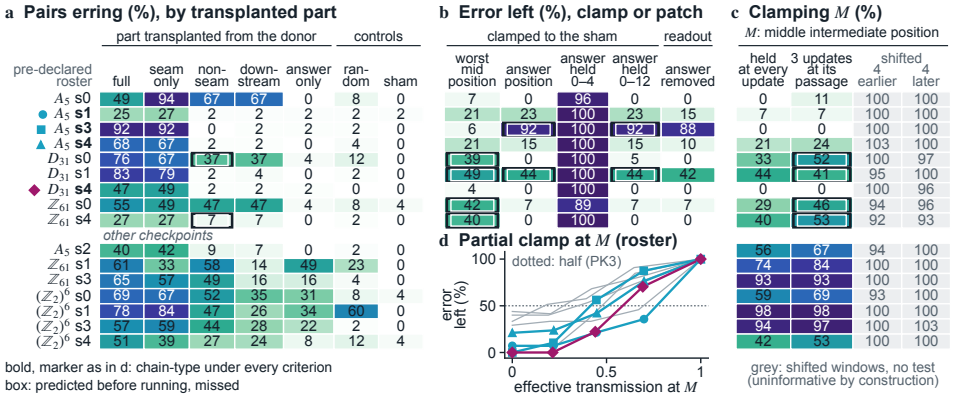


Figure 5: **The difference starts at the edited operands and can be blocked in transit.** Length 8; rows: task, seed (s0–s4); roster: edit-ordered checkpoints; bold ("chain-type"): the 4 meeting every edit-order criterion with room to spare; boxes: missed pre-set predictions (Appendix G.3). **(a)** Seam: the edited operands; non-seam: the rest; downstream: intermediate positions $3..L$; random: a same-size random whole-state displacement (worst of four). **(b)** Clamps to the untouched run (sham) vs. no clamp (operand-only transplant for mid positions, else full); worst mid position: the one leaving most error; answer held 0–4 or 0–12, then released; readout: answer position's difference removed. **(c, d)** Clamps of M in time and in strength (effective transmission: share of its difference let through; dotted: half the unclamped error, the pass threshold of prediction PK3).

4.2 Delaying it midway delays the error

We delay the difference at the middle position M (position 5 at length 8): each update, M receives the untouched run's state plus its own operand-only difference from k updates earlier. For $k{=}2$ and 4, 80–85% of pairs erring in both runs shift their first error by $k{\pm}1$ (pooled over checkpoints, at each length), and the median per-pair shift is within $k{\pm}1$ in 8 of 9 checkpoints (9 at length 6), while error rates change by $\leq$0.11 (0.14 at length 6): the delay moves errors in time (Figure 4d; $\leq$1 update in 6 of the 7 other checkpoints). Selection on error times alone cannot produce this shift.

Blocking agrees. Holding all intermediate positions at the untouched run removes most operand-only error in all 9 checkpoints at both lengths (Appendix G.4); each alone removes $\geq$75% in 5 (7 at length 6; Figure 5b). A three-update clamp of M at its peak difference in the pair's own run removes $\geq$60% in 5 and 6 of 9 (lengths 8, 6; Figure 5c); in the 4 most clearly edit-ordered, it removes 0.76–1.00 at the best window but 0.04–0.43 two updates later: the difference passes M within a few updates. An edit's *transit time* is the number of updates its difference needs to reach the readout via the intermediate positions; before it, a correct or stable readout can still miss the edit (Section 6).

4.3 The readout takes it from the answer position, upstream or both

Read with the untouched run's other positions, the transplant's answer position reproduces most errors in 4 of the 9 checkpoints (answer route: its state crosses the readout boundary), and its upstream positions in A_5 s3 (upstream route: they move the boundary across the answer position's state; Figure 4b,c); elsewhere, neither reproduces much over half (Appendix G.5). In A_5 s3 the delay shifts first errors by a median of exactly k: there, that route also waits for the transit. **Takeaway for patching and reuse.** Score patches per update; count readouts toward a stop after the transit time.

5 Error switches stay single under sixteen-fold refinement

Transient chaos (chaos that eventually settles; Altmann et al., 2013) is the natural rival account of late errors. Lai et al. (2026) report it in looped reasoning models: initial states labelled by convergence time form basins whose boundaries become fractal on harder tasks (Figure 6: their Countdown setting). A fractal boundary gains switches when refined; Table 1 tests this and five others.

	Chaos signature	Test (checkpoints) and result	Reading
1	Small differences change the outcome	All 16: random kicks of $9.7 \times 10^{-3}\,\|v\|$ change the transplant's lags-1–8 readouts in 0.41% of kicks; reaching the median error onset ($0.463\,\|v\|$) in 8 updates needs growth $\geq$0.48 per update, above every rate we estimated ($\leq$0.42)	Not discriminating: kicks too small
2	Growth in a moving frame (necessary)	Edit-ordered, pooled: the growth rate is negative at every frame speed from 0 to 1.5 positions per update in every resample (-0.087 per update at speed 1)	Decays when pooled; 2 checkpoints above zero uncorrected
3	Growth over a finite window	All 16, sampled untouched states: the 8-update map expands along its top direction at every state in 12 of 16 checkpoints (a lower-bound estimate)	Present; also arises without chaos (Hennequin et al., 2012)
4	Error times scatter, whatever the edit	Edit-ordered: the error peak moves by -1.21 to -0.57 updates per position of the edit	Not evidence: used to select these checkpoints
5	Refinement adds switches	All 16, stratified sample: 16-fold refinement leaves each of the 286 coarse error switches one fine switch ($\rho = 1.000$; synthetic dimension 1.2: $\rho = 1.18$)	Switches stay single; timing undecided
6	Wrong outputs are near-misses	Edit-ordered, both lengths, operand-only transplants: first wrong labels are products with one donor operand above chance in 1 of 36 cells (Bonferroni)	Rare; not specific to chaos

Table 1: **Six signatures of transient chaos, tested.** Length 8 and both-correct pairs unless stated; details in Appendix G.10. v: transplant minus untouched state h_S at insertion; ρ: fine per coarse switch. Rows 2–3 are measured around the untouched run.

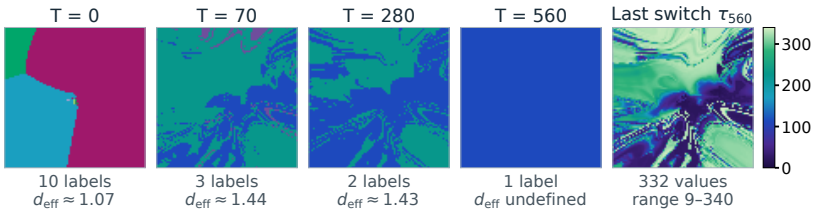


Figure 6: **Fate maps where transient chaos was reported** (Lai et al., 2026): our maps of a public looped Countdown model (Prairie et al., 2026), 513^2 initial states on a plane. Panels 1–4: label after T updates; 5: last change by T=560 (τ_{560}). d_{eff}: a fit-range-dependent grid exponent (Appendix S).

One fine switch per coarse switch. We label states on the line $h_S + s\,v$ ($s \in [-1, 2]$, step $0.01\,\|v\|$) by whether the readout errs in lags 1–8 and refine every switch 16-fold, the only level at least 100 times above round-off. In a stratified sample of all 16 checkpoints, each of the 286 coarse switches holds one fine switch ($\rho = 1.000$; 127/127 in the edit-ordered checkpoints). The refinement detects synthetic fractal boundaries of dimension $\geq$1.2: on transects of planar ones of dimension 1.2, matched in resolution and switches, it splits a median of 24 of the 286, and ρ's interval excludes 1 in 1000/1000 replicates (control and criterion post hoc; Appendix G.10).

Timing boundaries (first-error lag; last label change, the analogue of their convergence time) gain a few switches ($\rho = 1.01, 1.07$), which one refinement level cannot tell apart from the nested level sets of an integer-valued time, so they remain undecided. Growth over a finite window is present (row 3), but along the transplant's own run the whole difference stays below its size at insertion at every lag 1–20 in 97.0% of 830 both-correct pairs (length 8). We study the answer after a transplant in a task without trial and error, and Lai et al. study convergence time, so both results can hold.

A learned delay line. The readout's first-order response to the transplant's difference at the edited operands (the change in its margin at lag t) can stay well below its peak at first: its largest value over lags 1–3 is 0.01–0.36 of that peak in the 9 edit-ordered checkpoints but 0.71–1.00 in the other 7, which respond early, as causal attention allows. The largest response to a unit displacement d positions before the answer position, read d updates later, changes by under 3% per position in 6 of 9, so in these a displacement reaches the readout with about the same strength from any distance. Late errors, their removal by a midway clamp and their exact delay need no chaos: a linear recurrence in which each update moves every position's state one position toward the answer position and scales it by $0 < \alpha < 1$ produces all three (Appendix G.11). Proposition 2, in the same appendix, gives sufficient conditions for delayed arrival and exact delays, which the trained checkpoints need not meet. Under the delay of Section 4.2, the median per-pair first-error shift equals k for k=2 and 4 in 6 of 9 at length 8 and 8 of 9 at length 6.

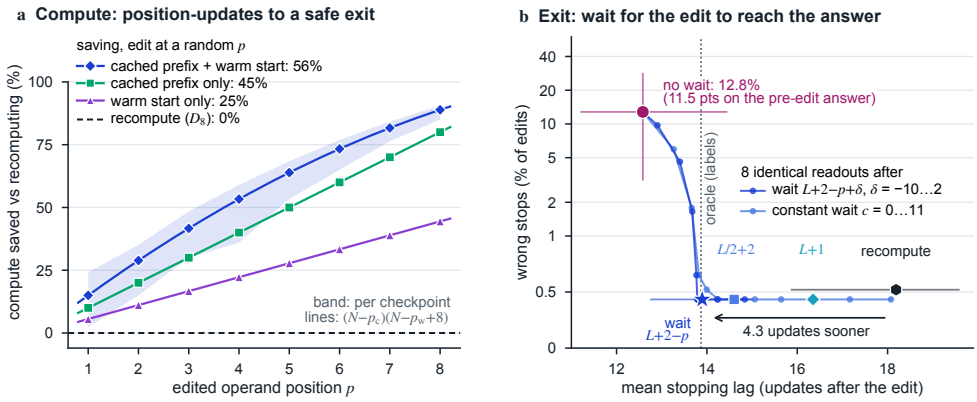


Figure 7: **Edit-aware reuse saves position-updates without adding wrong stops** (edit-ordered checkpoints, $L{=}8$, answer-changing edits at operand position p). **a** Saving against recomputing from h_0 with the eight-repeat exit (D_8): pooled medians (markers) and the recipe's per-checkpoint medians (band); lines: $1-(N{-}p_c)(N{-}p_w{+}8)/[N(N{+}8)]$, $N{=}L{+}2$, with p_c (p_w) $=p$ if the prefix is cached (the rest warm-started, waiting $L{+}2{-}p$), else 0. **b** Mean wrong stops and lags of eight-repeat exits on the warm start counted after a constant c (marked: $L/2{+}2$, $L{+}1$) or $L{+}2{-}p{+}\delta$ updates; bars: per-checkpoint ranges; oracle: end of the first eight correct readouts (needs labels).

Takeaway for dynamics analysts. Before attributing late errors to chaos, check whether error time tracks the edit and calibrate boundary refinement with a positive control.

6 An edit-aware recipe for reusing latent states

A prompt edit at operand position p need not restart the recurrence from h_0. On the 9 edit-ordered checkpoints, three rules make reuse cheaper than recomputing without adding wrong stops (Figure 7). **R1: read the positions left of p from a per-update cache of the unedited run.** Causal attention leaves them unchanged by the edit (Section 2), so R1 is exact prefix caching. **R2: start positions p to $L{+}1$ from their cached states** instead of h_0. **R3: count readouts toward the eight-repeat exit only from $L{+}2{-}p$ updates after the edit,** a wait for the edit's transit time (Section 4.2).

The warm start needs the wait. Counted from the edit, eight identical readouts stop 12.8% of warm starts on a wrong answer at $L{=}8$ (1.7% at $L{=}6$), 11.5 points of them on the pre-edit answer, which the readout keeps until the edit's difference reaches it. With the wait, 0.43% stop wrongly (0.00% at $L{=}6$), none on the pre-edit answer, against 0.52% (0.00%) for recomputing, and in no checkpoint more often; the exit comes 4.3 updates sooner (mean lag 13.9 vs. 18.2). Constant waits of $L/2{+}2$ and $L{+}1$ are as safe but exit after 14.6 and 16.4 updates, so the wait matters more than its dependence on p. In another model, first check with a small edit sweep (Section 2) that errors come later for edits farther from the answer; outside the edit-ordered checkpoints, the recipe can stop wrongly more often than recomputing (Appendix G.8).

Recomputing runs all $N{=}L{+}2$ positions for a median $N{+}8$ updates. The recipe runs $N{-}p$ positions (R1) and exits a median $N{-}p{+}8$ updates after the edit (R2 and R3), so, not counting the cache, it costs $(N{-}p)(N{-}p{+}8)$ position-updates instead of $N(N{+}8)$; on these checkpoints the product equals the pooled median at every p and both lengths. For uniform p, the recipe saves 56% of recomputing's position-updates at $L{=}8$ (46–61% per checkpoint; 54% at $L{=}6$), or 55% with the constant wait $L/2{+}2$. Plain prefix caching, R1 alone, saves 45% but exits when recomputing does, 1.0–6.1 updates after the recipe per checkpoint; warm starts alone save 25%.

R4: judge a replaced state by its continuation. Compare its next $L{+}1$ readouts with the untouched run's (for a both-correct state: if that run stays correct, a pass places the state in $\mathcal{R}_y^{L+1}$ of Section 4) and say whether each donor keeps or changes the answer (Section 3.2). The horizon is calibrated on these data: in every task and length, $\geq$97.4% of both-correct answer-preserving transplants that later err first do so by lag $L{+}1$. On these transplants in Experiment II, the check flags 56.6% at

Table 2: **Checks for patching, reusing and stopping recurrent states** (L=8 unless stated).

Situation and check	What breaks without it	Evidence	Cost
Patching study. Score against the untouched run; say whether donors keep the answer.	A fixed mean settles 0.67–0.82 updates before answer-changing donors but 2.86–5.57 after answer-preserving ones (by task).	Frozen; Exp. II; Fig. 3	One run.
Judging a replaced state (R4). Compare its next L+1 readouts with the untouched run's.	53.36% of both-correct answer-preserving transplants err within eight updates (sham: 0.84%), as do 47.0% of those Lemma 1 certifies.	Descriptive; Exp. II; Fig. 2	$2(L+1)$ updates.
Prompt edit (R1–R3). Reuse the prefix, warm-start the rest; count stable outputs from lag $L+2-p$.	Eight-repeat exits stop 12.8% of warm starts wrongly after answer-changing edits, 11.5 points on the pre-edit answer.	Exploratory; Fig. 7	A cache; the wait.
Label-free exit. Test on starts stable at a wrong answer, at a stated wrong-stop budget.	Eight-repeat exits stop 6.35–35.20% of answer-changing transplants wrongly (1.76–3.52% at L=6; by task).	Frozen contrasts; Exp. II; Fig. 3	Later exits.

L=8, 99.6% of which are correct again at lag 35, and 1.72% of those it passes err later. It needs a reference run, so it audits a method on sample states.

7 Related work, limitations and conclusion

Related work. Patching results depend on the replacement and metric (Zhang & Nanda, 2024; Heimersheim & Nanda, 2024); a patch can succeed through a normally dormant pathway (Makelov et al., 2024), and patching a state into another prompt decodes what it carries (Ghandeharioun et al., 2024). Adaptive computation stops by a learned halting unit (Graves, 2016) or once a confidence measure passes a threshold (Schuster et al., 2022); path independence concerns where iteration ends (Anil et al., 2022). In looped and recurrent-depth models, the latent state is read at every update, reused as a warm start and checked for an exit (Geiping et al., 2025); decoding such states finds limited evidence of interpretable latent reasoning (Lu et al., 2025), and transient chaos with fractal convergence-time basins has been reported (Lai et al., 2026). We ask instead when a replaced state should be read: its readout can be right at insertion and at the end yet wrong in between.

Limitations. Our evidence covers one looped architecture, four finite-group tasks, the 16 checkpoints that pass our accuracy gate and two lengths; effect sizes are model-specific. The mechanism is traced in the 9 edit-ordered checkpoints, a set that depends on its criterion (4 meet every one we tried with room to spare); the positions the readout uses differ by checkpoint, and the trace does not show what the difference encodes or which pairs err (Appendix G.5). Section 5 excludes, at the resolution it tests, fractal error boundaries at least as rough as its positive control, measures growth rates along the untouched run only, and leaves timing boundaries undecided (Appendix G.10). The recipe's payoff is measured on the prompts used to select the edit-ordered checkpoints, with new edits, counting only position-updates and assuming the unedited run is cached up to the recipe's exit (Appendix G.8); the continuation check's horizon is calibrated on these data, and other checkpoints and models, including language models, need their own edit sweep and audit. Intervals resample prompts with training seeds fixed (3–5 per task).

Conclusion. In looped transformers trained on finite-group composition, an answer-preserving transplant decoding the correct answer turns wrong within several updates far more often than the untouched run, then almost always recovers; in edit-ordered checkpoints its difference reaches the readout mainly via intermediate positions, and errors arrive with it; refining their boundary adds no switches at the resolution we test. Where errors track edit distance, reuse after an edit should count stable outputs only after this arrival; in these models, any replaced state should be judged by its next few updates. Deciding whether a state lies in a continuation set $\mathcal{R}_y^H$ before running its continuation remains open.

REFERENCES

Eduardo G. Altmann, Jefferson S. E. Portela, and Tamás Tél. Leaking chaotic systems. *Reviews of Modern Physics*, 85:869–918, 2013. doi: 10.1103/RevModPhys.85.869.

Cem Anil, Ashwini Pokle, Kaiqu Liang, Johannes Treutlein, Yuhuai Wu, Shaojie Bai, J. Zico Kolter, and Roger B. Grosse. Path independent equilibrium models can better exploit test-time computation. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2211.09961.

Alvar Daza, Alexandre Wagemakers, Bertrand Georgeot, David Guéry-Odelin, and Miguel A. F. Sanjuán. Basin entropy: a new tool to analyze uncertainty in dynamical systems. *Scientific Reports*, 6:31416, 2016. doi: 10.1038/srep31416.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. Universal transformers. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=HyzdRiR9Y7.

Robert J. Deissler and Kunihiko Kaneko. Velocity-dependent Lyapunov exponents as a measure of chaos for open-flow systems. *Physics Letters A*, 119(8):397–402, 1987. doi: 10.1016/0375-9601(87)90581-0.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2502.05171.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15466–15490, 2024. URL https://proceedings.mlr.press/v235/ghandeharioun24a.html.

Elmer G. Gilbert and K. T. Tan. Linear systems with state and control constraints: The theory and application of maximal output admissible sets. *IEEE Transactions on Automatic Control*, 36(9):1008–1020, 1991. doi: 10.1109/9.83532.

Alex Graves. Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*, 2016. URL https://arxiv.org/abs/1603.08983.

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching. *arXiv preprint arXiv:2404.15255*, 2024. URL https://arxiv.org/abs/2404.15255.

Guillaume Hennequin, Tim P. Vogels, and Wulfram Gerstner. Non-normal amplification in random balanced neuronal networks. *Physical Review E*, 86:011909, 2012. doi: 10.1103/PhysRevE.86.011909.

Robert Hermann and Arthur J. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, 1977. doi: 10.1109/TAC.1977.1101601.

Jeffrey Lai, Anthony Bao, John Quinn, and William Gilpin. Fractal basins trap latent reasoning. *arXiv preprint arXiv:2609.04963*, 2026. URL https://arxiv.org/abs/2609.04963.

Bingbin Liu, Jordan T. Ash, Surbhi Goel, Akshay Krishnamurthy, and Cyril Zhang. Transformers learn shortcuts to automata. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.10749.

Wenquan Lu, Yuechuan Yang, Kyle Lee, Yanshu Li, and Enqi Liu. Latent chain-of-thought? decoding the depth-recurrent transformer. *arXiv preprint arXiv:2507.02199*, 2025. URL https://arxiv.org/abs/2507.02199v2.

Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. Is this the subspace you are looking for? An interpretability illusion for subspace activation patching. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2311.17030.

Steven W. McDonald, Celso Grebogi, Edward Ott, and James A. Yorke. Fractal basin boundaries. *Physica D: Nonlinear Phenomena*, 17(2):125–153, 1985. doi: 10.1016/0167-2789(85)90001-6.

William Merrill, Jackson Petty, and Ashish Sabharwal. The illusion of state in state-space models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35492–35506, 2024. URL https://proceedings.mlr.press/v235/merrill24a.html.

Hayden Prairie, Zachary Novack, Taylor Berg-Kirkpatrick, and Daniel Y. Fu. Parcae: Scaling laws for stable looped language models. *arXiv preprint arXiv:2604.12946*, 2026. URL https://arxiv.org/abs/2604.12946.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2207.07061.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, 2019. URL https://arxiv.org/abs/1910.07467.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.16042.

APPENDIX: ADMINISTRATIVE DISCLOSURES

Reproducibility statement. Both independent-pool experiments froze inputs, code, checkpoint hashes and analyses before measurement. Full per-pair logits and event flags support reconstruction of every outcome. Independent implementations verified the block aggregates and simultaneous intervals; Experiment I's timing/stopping extensions are post-hoc; Experiment II froze their fresh-input replication before measurement (not externally registered). Its new insertion and sham reanalyses remain post-hoc. Appendix U identifies plans, archives, failures and verification scope. Section 4's intervals resample prompts (33–63 pairs per checkpoint and length) with training seeds fixed (3–5 per task). Script predictions for the exploratory analyses follow earlier outputs (none blind; Appendix G.3). Specification–code mismatches remain open (Appendix G.13).

AI use statement. Generative AI assisted code, analysis, mathematical review, literature checks and writing. Automated cross-checks are not independent human replications. LLM coding agents of one model family (Claude) implemented and checked every exploratory analysis of Sections 4–6, so their errors may be shared; the order of their pre-stated predictions and outputs rests on session records, not external time stamps (Appendix G.3). The authors are responsible for the final scientific claims and submission.

GUIDE TO THE APPENDICES

Evidence levels. We use the three levels of Section 2. *Frozen:* contrasts frozen by hash before measurement (not externally registered) in Experiments I and II and in the Countdown test (Appendix W). *Descriptive:* post hoc analyses of those saved runs. *Exploratory:* the update-40 control of Section 3.1, the mechanism and dynamics analyses of Sections 4 and 5 and the reuse analyses of Section 6. The appendices tag them [R], [D] and [X]. Where a script stated a prediction before its run, the appendix says so and scores it by its rule as written (Appendix G.3); every other analysis written after the data existed is called post hoc.

Exploratory status of Sections 4, 5 and 6. Their intervals mostly resample contexts or pairs within checkpoints (33–63 pairs per checkpoint and length in Section 4; 16 contexts per checkpoint in the reuse experiment), not training seeds (3–5 per task). Script predictions were written after earlier outputs had been seen, so none is blind, and their order relative to the outputs rests on file times and session records kept by the same pipeline, not on external time stamps (Appendix G.3). Only LLM coding agents of one model family (Claude) implemented and checked these analyses, so their errors may be shared. Discrepancies between specifications and code that remain open are listed in Appendix G.13.

Further scope. Donor norms are recorded, not matched, so answer congruence is confounded with how compatible an edit is with the recipient's state. Transplants enter at the insertion time; the reuse experiment also uses three times that update, and a control inserts at update 40 (Appendix G.9). Three questions about the mechanism stay open: what the carried difference encodes (script prediction EM5 not met), why it moves from position to position although the answer position attends to every operand from the first update, and which pairs err (the answer position's fixed point also turns wrong in 0.54 of the seam-only pairs that never err, pooled over the edit-ordered checkpoints at $L{=}8$; Appendix G.5). In a Countdown model a small kick at any operand position reaches the last position within one update (492/492 cases; Appendix G.10), so the route through intermediate positions need not carry over to other tasks.

Names. Table 3 maps the main text's names to the shorter ones of the appendices, each also defined at first use.

Main text	Appendices
accuracy gate	capability gate
untouched run (the recipient's own state, re-inserted)	sham, h_S
the two edited operands (positions 1 and 2)	seam
edit-ordered checkpoints: errors come later for farther edits; fixed before the clamp and patch runs B and C (A_5 s0, s1, s3, s4; D_{31} s0, s1, s4; $\mathbb{Z}_{61}$ s0, s4)	roster ("pre-declared chain roster" in tables)
the four that meet every edit-order criterion with room to spare, a sensitivity set fixed after the first mechanism runs (A_5 s1, s3, s4; D_{31} s4)	robust four; bold, "chain-type", in Figure 5
insertion time: first of three consecutive updates with at least 80% calibration accuracy	t_{80}
settling time: end of the first run of eight correct readouts	C_8
eight-repeat exit: end of the first run of eight identical readouts, correct or not	D_8
first correct lag; wrong readouts in the eight updates after it; start of the last correct run up to lag 27	A_{27}; W_8; S_{27}
the middle intermediate position	M
the answer position's fixed point with the upstream positions held	effective map, ℓ^*
the reuse experiment	E3 script, predictions E1–E5 (Appendix G.7)
script predictions	P1–P3, Pa–Pd, Pb′, E1–E5, EM1–EM5, PK1–PK3, X1a–X1c, GA0–GA2, GC1–GC3, R3a–R3b, X2a–X2b (Appendix G.3)

Table 3: **Names in the main text and in the appendices.**

Where the evidence for each claim is. Table 4 lists each main-text claim with its evidence level and the appendices holding its methods and complete results.

Claim (main text)	Evidence	Appendices
In every accuracy-gated checkpoint, answer-preserving transplants that decode correctly turn wrong within eight updates far more often than the untouched run, then recover (Section 3.1)	descriptive, on Experiment II's saved runs; update-40 control exploratory	D, C, B, G.9
Whether a replacement helps depends on the comparator and on the donor type (Section 3.2)	frozen contrasts; the untouched run descriptive	B (Experiment II), E and A (Experiment I)
A stable output can repeat the donor's answer (Section 3.3)	frozen; rule variants descriptive	C, D
In the edit-ordered checkpoints, the difference passes mainly through intermediate positions to the positions the readout uses (Section 4)	exploratory; script predictions scored	G.1–G.4, G.5, F.1, G.6, G.9; verdicts G.3
Refining the error boundary sixteen-fold adds no switches; delay-line measurements; late errors need no chaos (Section 5; the example and Proposition 2 are in Appendix G.11)	exploratory; positive control post hoc; proposition proved; band test pre-fixed, rejected	G.10, G.11, S
The edit-aware recipe for reusing latent states (Section 6)	exploratory; post hoc on the reuse experiment and Experiment II	G.7, G.8
Accounts we tested and did not support; open specification–code discrepancies	exploratory	G.12, G.13
A frozen transplant test in a Countdown model (appendix only)	frozen	W

Table 4: **Claims, evidence and appendices.** Letters refer to the appendices below.

A POST-HOC TIMING CONTRASTS RETAIN INDEPENDENT-POOL UNCERTAINTY

These analyses were specified after inspecting the earlier task means and are explicitly post-hoc. No model was rerun. The 24-cell family is four tasks by three metrics (A_{27}, C_8, W_8) by two contrasts (fixed mean minus donor; fixed mean plus noise minus fixed mean). A separate 16-cell family compares native and donor for A_{27}, S_{27}, C_8, W_8. Both use 50,000 block-bootstrap draws, resampling each task's 32 blocks independently across tasks and synchronously across the checkpoints and contrasts within that task. Centered bootstrap task means are divided by the observed block standard error; the maximum absolute statistic calibrates a two-sided familywise 95% interval. Critical values are 2.97477 and 2.78124, respectively. Seed identifiers for resampling are 20260923 and 20260924. Inference remains conditional on the frozen capable checkpoint roster. A zero standard error yields unavailable inference. These families do not replace the registered 20- and 440-cell results.

A.1 ADDING NOISE AT FIXED POOL SCALE

Task	Metric	Effect [simultaneous 95% interval]
$\mathbb{Z}_{61}$	A_{27}	+0.01 [-0.18, +0.21]
$\mathbb{Z}_{61}$	C_8	+0.07 [-0.01, +0.15]
$\mathbb{Z}_{61}$	W_8	+0.04 [-0.12, +0.20]
$(\mathbb{Z}_2)^6$	A_{27}	-0.08 [-0.24, +0.09]
$(\mathbb{Z}_2)^6$	C_8	+0.10 [-0.09, +0.29]
$(\mathbb{Z}_2)^6$	W_8	+0.03 [-0.10, +0.16]
D_{31}	A_{27}	-0.01 [-0.21, +0.19]
D_{31}	C_8	-0.03 [-0.13, +0.07]
D_{31}	W_8	+0.02 [-0.15, +0.19]
A_5	A_{27}	+0.04 [-0.12, +0.19]
A_5	C_8	+0.18 [+0.10, +0.27]
A_5	W_8	+0.11 [-0.01, +0.24]

A.2 NATIVE MINUS COMPUTED DONOR

Task	Metric	Effect [simultaneous 95% interval]
$\mathbb{Z}_{61}$	A_{27}	-0.81 [-1.13, -0.48]
$\mathbb{Z}_{61}$	S_{27}	+0.44 [+0.29, +0.59]
$\mathbb{Z}_{61}$	C_8	+0.46 [+0.31, +0.61]
$\mathbb{Z}_{61}$	W_8	+0.90 [+0.70, +1.10]
$(\mathbb{Z}_2)^6$	A_{27}	+0.10 [-0.27, +0.48]
$(\mathbb{Z}_2)^6$	S_{27}	+1.33 [+1.09, +1.56]
$(\mathbb{Z}_2)^6$	C_8	+1.31 [+1.07, +1.55]
$(\mathbb{Z}_2)^6$	W_8	+0.82 [+0.62, +1.03]
D_{31}	A_{27}	-0.23 [-0.56, +0.10]
D_{31}	S_{27}	+0.59 [+0.43, +0.74]
D_{31}	C_8	+0.56 [+0.40, +0.73]
D_{31}	W_8	+0.58 [+0.38, +0.79]
A_5	A_{27}	+0.83 [+0.58, +1.09]
A_5	S_{27}	+1.75 [+1.65, +1.86]
A_5	C_8	+1.75 [+1.65, +1.86]
A_5	W_8	+0.64 [+0.49, +0.78]

Task	Fixed mean only − donor	Simultaneous 95% interval
Z61	-0.680	$[-0.88, -0.48]$
$(\mathbb{Z}_2)^6$	-0.694	$[-1.04, -0.35]$
D31	-0.767	$[-1.02, -0.51]$
A5	-0.866	$[-1.01, -0.72]$

Table 5: **The same statistical state helps different evaluation problems.** Difference in capped correct-suffix latency S_{27}, in updates. The deterministic mean uses construction-pool median norm and is constant within each block. All four secondary-family intervals favor it over donor. This control removes both evaluation-donor coordinates and evaluation-donor scale.

Task	First hit A_{27}	Completion C_8	Post-hit errors W_8
$\mathbb{Z}_{61}$	-1.94 [-2.24, -1.64]	-0.71 [-0.83, -0.58]	+0.85 [+0.65, +1.06]
$(\mathbb{Z}_2)^6$	-1.77 [-2.09, -1.45]	-0.71 [-0.98, -0.43]	+0.67 [+0.51, +0.84]
D_{31}	-1.38 [-1.67, -1.09]	-0.79 [-1.01, -0.58]	+0.39 [+0.25, +0.53]
A_5	-1.43 [-1.67, -1.18]	-0.87 [-0.98, -0.75]	+0.35 [+0.23, +0.48]

Table 6: **No injected noise is needed for the temporal reversal.** Fixed, pool-scaled mean minus donor, in updates for A_{27}, C_8 and counts for W_8. Every interval excludes zero in the displayed direction. These are post-hoc 95% simultaneous intervals over 24 task-level timing/noise contrasts, distinct from the registered families.

	Wrong stop (% of all pairs)		Fixed mean minus donor	
Task	Donor	Fixed mean	Wrong stop (pp)	Stop lag (updates)
$\mathbb{Z}_{61}$	15.19	0.39	-14.8 [-16.5, -13.1]	+0.80 [+0.57, +1.03]
$(\mathbb{Z}_2)^6$	6.59	0.49	-6.1 [-7.9, -4.3]	+0.04 [-0.18, +0.26]
D_{31}	29.82	1.30	-28.5 [-31.4, -25.6]	+2.20 [+1.84, +2.56]
A_5	30.98	0.47	-30.5 [-32.8, -28.2]	+2.19 [+1.94, +2.45]

Table 7: **Repeated predictions can stabilize on the wrong answer.** Stop at the first eight identical predicted tokens, including the insertion-time readout. This rule uses no target label. Wrong-stop probabilities use all pairs, not only those that stop. Stop lag is capped at 35; XOR has 3/2,048 donor and 4/2,048 fixed-mean non-stops, with none in the other displayed cells. Intervals are from a separate post-hoc 12-cell simultaneous family.

A.3 Reconstructing the readout and a label-free stopping rule

The analysis verifies all 8,192 pair files against their block manifests and reconstructs full-vocabulary argmax at each of 36 lags in all seven conditions (2,064,384 logit vectors). Group labels map to vocabulary indices with offset two. Insertion margins are the recipient-token logit minus the largest other vocabulary logit. Their descriptive means use the same sixteen-checkpoint cohort as the new timing contrasts, not the older 1,024-pair archive.

The stopping addendum was specified after viewing descriptive stopping rates and is post-hoc. For fixed mean minus donor it jointly analyzes four tasks by three outcomes: unconditional wrong-stop probability, non-stop probability, and stopping lag capped at 35. It uses 50,000 block-bootstrap draws with seed 20260925 and the same roster/block weighting as the timing analysis; max-t critical value is 2.66516. Three non-stop contrasts have zero block variance and receive no inferential interval, rather than being declared equivalent. XOR's non-stop difference is 0.049 percentage points $[-0.246, 0.344]$. Conditional wrong-stop rates and their denominators are retained separately in the analysis JSON. This family is separate from both registered inference and the post-hoc temporal families. No multiplicity guarantee across these separately declared families is asserted.

The rule starts counting at insertion. It intentionally tests whether immediate agreement suffices for stopping, rather than using an oracle first-correct time as a burn-in. A donor can therefore complete a wrong-label run before ever reaching the recipient answer. The new answer-congruence experiment registers this outcome before collecting new continuations.

B Completed matched-donor experiment: registration, results and exceptions

Experiment II is a completed, separate preregistered test of the answer-conflict alternative and of the temporal pattern first observed in Experiment I. Inputs, code, checkpoints and analysis were frozen before new outcomes. Both answer-preserving and answer-changing donors alter the same first two operand positions as in Eq. (1); both positions must differ from the original, and their second operand is shared. The construction is valid for noncommutative products because order is retained. Donor norms are recorded rather than equalized.

Independent blocks and calibration. Each task/length has 32 blocks of 64 construction contexts and eight evaluation contexts, with two matched edits per evaluation context. The construction pool has 64 preserving and 64 changing states from independent problems. Construction, evaluation, calibration and preflight sequences, including transformed operands, are screened against one another and recorded historical inputs. This is not a claim of exhaustive separation from unrecorded training histories. Blocks are shared across checkpoints within each task/length, giving 2,048 unique evaluation contexts, 4,096 matched edits and 16,384 checkpoint–pair records across both lengths.

All sixteen fixed capable checkpoints completed both lengths. Length-six eligibility required update-27 accuracy at least 0.90 and a sustained t_{80}, measured on 512 independent calibration contexts per checkpoint. A5/s2 scored 511/512 and the other fifteen 512/512. Length six is within the original curriculum 1–8, not out-of-training-length generalization. The four previous failures (Z61/s2, XOR/s2, D31/s2,s3) remain failures; there was no retraining or replacement.

Frozen predictions and outcomes. P1 predicted shorter S_{27}/C_8 and fewer wrong stops from preserving than changing donors. All four task intervals support these directions at both lengths. P2 stipulated that a mean beating preserving donors would exclude conflict removal alone as an explanation within the design. This criterion is not met: all eight task/length contrasts on both S_{27} and C_8 instead favor preserving donors. This does not identify conflict removal as the sole mechanism. P3 predicted earlier A_{27}/C_8, higher W_8, and fewer wrong stops for the fixed mean versus changing donors. All four task intervals at each length support these directions.

Simultaneous families and non-events. Each length has its own 48-cell primary family (four tasks, three contrasts and A_{27}, S_{27}, C_8, W_8) and 36-cell stopping family (wrong-stop probability, non-stop probability and capped stop lag). Fifty thousand synchronized block-bootstrap draws yield primary critical values 3.13064293 (length eight) and 3.07960053 (length six); stopping values are

3.0 for each length. There is no cross-family or cross-length familywise guarantee. Twenty-one zero-standard-error cells retain unavailable intervals, not equivalence. Inference conditions on the fixed capable roster. Checkpoint differences below are descriptive and do not estimate a seed population.

Events are reconstructed from full 128-token logits at lags 0–35 for all six conditions. A/S non-events cap at 27, C/D at 35, with separate flags; W_8 assigns eight to non-hits. Wrong-stop probabilities use all pairs as denominator. At length eight the largest C_8 noncompletion rate among preserving/mean/changing task cells is 0.879% (XOR/changing); length six has the same maximum. Tables retain the capped estimands rather than treating them as uncensored means.

Table 8: **The mean wins against changing donors and loses against preserving donors.** C_8 contrasts by task and sequence length. Entries are estimate [simultaneous 95% CI] in updates; contrasts are left minus right, so negative values favor the fixed mean. Intervals are simultaneous within a separate registered 48-cell primary family for each sequence length (50,000 max-$|t|$ draws).

Length	Task	Mean − changing donor	Mean − preserving donor
8	Z_{61}	-0.745 [-0.914, -0.576]	+4.500 [+4.267, +4.734]
	$(\mathbb{Z}_2)^6$	-0.667 [-0.981, -0.353]	+2.863 [+2.488, +3.239]
	D_{31}	-0.773 [-1.000, -0.546]	+5.570 [+5.226, +5.915]
	A_5	-0.822 [-0.917, -0.726]	+5.563 [+5.257, +5.869]
6	Z_{61}	-0.860 [-1.003, -0.717]	+2.931 [+2.661, +3.201]
	$(\mathbb{Z}_2)^6$	-1.208 [-1.413, -1.003]	+1.624 [+1.285, +1.963]
	D_{31}	-0.917 [-1.111, -0.722]	+2.097 [+1.664, +2.530]
	A_5	-0.908 [-0.988, -0.828]	+2.916 [+2.683, +3.149]

Sham and initialization anchor the comparison. At length eight, sham C_8 means range from 7.21 to 7.36, native resets from 15.55 to 18.95, fixed means from 14.32 to 17.38, and pool-scaled isotropic starts from 16.73 to 19.51. These are descriptive baseline values, not additional registered hypothesis tests. Sham is a continuation at t_{80}, whereas native is a reset to h_0. The shorter sham times establish the relevant zero point: a useful replacement is not thereby faster than intact inference.

Technical exposure, acquisition and independent checks. One A5 context was exposed during a pre-freeze smoke test; both edits were retired and replaced by a separately seeded, nonoverlapping context before freezing. All 32 numerical and save/reload preflights passed. The frozen hashes were rechecked after execution. All 16,384 raw pair files, comprising 3,538,944 logit vectors, were retrieved with member hashes checked. Independent postprocessing reconstructed every label and event, the block aggregates, and all 168 interval entries; inferential quantities matched exactly and descriptive means within 7.11×10^{-15}. These checks establish provenance and arithmetic; they do not guarantee finite-sample interval coverage.

C Insertion readouts, sham contrasts and complete stopping outcomes

This explicitly post-hoc audit uses all 16,384 matched-study pair records and 98,304 condition trajectories. It reconstructs every initial target margin $\ell_y - \max_{j \neq y} \ell_j$, predicted label, A_{27}, C_8, W_8, and D_8 outcome directly from saved full-vocabulary logits. Every saved endpoint agrees; independent reconstruction of the task-average endpoints differs by at most 3.6×10^{-15}. Input, script and raw-member hashes accompany the compact per-pair audit. No new GPU measurement, filtering of primary pairs or alteration of registered criteria is involved.

C.1 Initial answer agreement and first-hit advantage

The descriptive summaries average checkpoints equally within each task. Each tuple reports *initial accuracy in percent; mean target margin; mean whole-state norm.* Within each task the six

Table 9: Temporal replication of first-hit time and post-hit errors. Entries are estimate [simultaneous 95% CI], left minus right. A27 is first correct-answer update; W8 is errors in the eight updates after first correctness. Each length uses its own registered 48-cell primary family.

Length	Task	Mean–changing A27	Mean–changing W8	Preserving–changing W8
8	Z_{61}	-1.895 [-2.223, -1.567]	+0.800 [+0.607, +0.994]	+0.984 [+0.824, +1.144]
8	$(\mathbb{Z}_2)^6$	-1.562 [-1.936, -1.189]	+0.567 [+0.391, +0.743]	+1.437 [+1.325, +1.550]
8	D_{31}	-1.408 [-1.694, -1.121]	+0.426 [+0.250, +0.603]	+1.174 [+1.052, +1.297]
8	A_5	-1.171 [-1.379, -0.964]	+0.245 [+0.117, +0.373]	+0.834 [+0.768, +0.901]
6	Z_{61}	-1.570 [-1.778, -1.363]	+0.453 [+0.348, +0.557]	+1.055 [+0.906, +1.203]
6	$(\mathbb{Z}_2)^6$	-1.731 [-2.033, -1.429]	+0.292 [+0.120, +0.464]	+1.451 [+1.305, +1.596]
6	D_{31}	-1.313 [-1.573, -1.052]	+0.270 [+0.133, +0.408]	+1.885 [+1.713, +2.057]
6	A_5	-1.191 [-1.353, -1.029]	+0.189 [+0.102, +0.275]	+1.075 [+0.998, +1.152]

Table 10: Unconditional D8 wrong-stop rate by task and length (percent of all pairs; descriptive).

Length	Task	Changing donor	Preserving donor	Fixed mean
8	Z_{61}	15.527	0.342	0.195
8	$(\mathbb{Z}_2)^6$	6.348	1.416	0.098
8	D_{31}	27.865	0.326	0.391
8	A_5	35.195	0.508	0.469
6	Z_{61}	3.516	0.049	0.000
6	$(\mathbb{Z}_2)^6$	2.637	0.977	0.000
6	D_{31}	2.539	0.391	0.651
6	A_5	1.758	0.000	0.000

tuples are, in order: *sham, preserving donor, changing donor, fixed mean, native reset, pool-scaled isotropic*. Similar average margins do not assert matched distributions or statistical equivalence. Norm averages are reported, not equalized.

Length 8.

- Z_{61}: (88.09; +6.252; 2024.90), (86.13; +6.257; 2020.94), (0.20; -25.595; 2022.94), (1.86; -4.184; 2004.68), (1.17; -10.266; 1.97), (1.56; -10.802; 2004.68).
- $(\mathbb{Z}_2)^6$: (87.01; +6.936; 2815.75), (88.67; +7.107; 2818.08), (0.29; -24.613; 2817.53), (1.56; -4.340; 2814.88), (0.98; -10.675; 1.97), (1.86; -13.711; 2814.88).
- D_{31}: (89.84; +5.640; 2236.26), (89.26; +5.681; 2235.19), (0.26; -23.115; 2236.19), (1.17; -2.698; 2236.94), (1.04; -9.198; 1.97), (1.56; -9.284; 2236.94).
- A_5: (86.88; +5.580; 1854.51), (86.45; +5.606; 1855.01), (0.20; -23.736; 1854.62), (0.78; -4.152; 1849.16), (1.48; -7.020; 1.97), (1.84; -8.738; 1849.16).

Length 6.

- Z_{61}: (89.26; +7.163; 1833.92), (90.19; +7.210; 1832.97), (0.20; -26.152; 1832.83), (1.27; -5.289; 1817.00), (1.66; -10.190; 1.76), (1.32; -10.695; 1817.00).
- $(\mathbb{Z}_2)^6$: (87.11; +7.405; 2589.77), (87.94; +7.556; 2587.78), (0.34; -25.322; 2588.87), (0.68; -5.185; 2582.33), (1.37; -10.372; 1.76), (1.51; -13.912; 2582.33).
- D_{31}: (85.29; +5.256; 1953.49), (84.44; +5.465; 1953.23), (0.26; -24.508; 1953.99), (2.08; -3.010; 1956.02), (2.86; -8.975; 1.76), (1.95; -9.148; 1956.02).
- A_5: (88.52; +5.839; 1654.96), (87.93; +5.771; 1654.31), (0.23; -24.569; 1654.20), (1.48; -4.017; 1648.54), (1.09; -7.149; 1.76), (1.25; -8.803; 1648.54).

Table 11: D8 wrong-stop contrasts in percentage points. Entries are estimate [simultaneous 95% CI], left minus right; negative values favor the left condition. Each length uses its separate registered 36-cell stopping family.

Length	Task	Preserving–changing	Mean–changing
8	Z_{61}	-15.19 [-17.62, -12.76]	-15.33 [-17.85, -12.81]
8	$(\mathbb{Z}_2)^6$	-4.93 [-6.80, -3.06]	-6.25 [-8.12, -4.38]
8	D_{31}	-27.54 [-30.08, -25.00]	-27.47 [-29.81, -25.14]
8	A_5	-34.69 [-37.67, -31.70]	-34.73 [-37.68, -31.78]
6	Z_{61}	-3.47 [-5.00, -1.94]	-3.52 [-5.05, -1.98]
6	$(\mathbb{Z}_2)^6$	-1.66 [-2.81, -0.51]	-2.64 [-3.68, -1.60]
6	D_{31}	-2.15 [-3.14, -1.16]	-1.89 [-3.09, -0.69]
6	A_5	-1.76 [-2.45, -1.07]	-1.76 [-2.45, -1.07]

Registered first-hit comparison. Preserving minus changing A_{27} estimates and simultaneous intervals, in the task order above, are: length 8: -7.422 [-7.611, -7.232], -7.617 [-7.885, -7.349], -10.561 [-10.718, -10.403], -9.704 [-9.802, -9.606].

length 6: -6.027 [-6.149, -5.906], -6.637 [-6.846, -6.428], -8.506 [-8.672, -8.340], -7.530 [-7.619, -7.442].

These intervals belong to each length's registered 48-cell family, not the new sham family. The large initial-correctness difference is part of the measured total contrast; it prevents interpreting the completion advantage as a post-hit acceleration.

C.2 Already-correct starts can subsequently diverge

Conditioning on both sham and preserving starts being correct at lag zero yields 6,582 pairs at length eight. In this same selected set, preserving continuations make at least one wrong prediction in lags 1–8 in 53.3576% of pairs, versus 0.8356% for sham; the mean numbers of wrong lags are 1.23169 and 0.01155. Length six gives 6,554 selected pairs, 56.2862% versus 0.9002%, and 1.41562 versus 0.01404 wrong lags. Counts pool the fixed checkpoint roster and correlated edits. They are descriptive, not independent sample sizes for binomial tests, seed-population estimates, or controlled direct effects. Selecting on post-intervention correctness can change the composition of the population.

C.3 Post-hoc sham families retain unselected pairs

For each length separately, four tasks, three contrasts (preserving, fixed mean, changing minus sham), and two outcomes (C_8, W_8) define a new 24-cell family. Each task has 32 block means, first averaging its checkpoints equally. Task-specific block indices are independently resampled, synchronously across all of that task's cells, in 50,000 centered max-$|t|$ bootstrap draws (seed 23092319). Critical values are 2.99410423 at length eight and 3.00021172 at length six. Zero-standard-error intervals remain unavailable. These are post-hoc simultaneous intervals conditional on the roster, with no joint guarantee across lengths or registered families.

Each entry is estimate [simultaneous 95% interval]. Each task lists C_8 then W_8 for preserving–sham, mean–sham and changing–sham, respectively. Time is in updates; W_8 is an error count.

Length 8.

- Z_{61}: $+2.571$ [+2.329, +2.812]; $+1.152$ [+0.997, +1.308]; $+7.071$ [+6.886, +7.257]; $+0.969$ [+0.796, +1.142]; $+7.816$ [+7.664, +7.968]; $+0.168$ [+0.113, +0.224].
- $(\mathbb{Z}_2)^6$: $+4.550$ [+4.201, +4.899]; $+1.720$ [+1.588, +1.852]; $+7.413$ [+7.255, +7.571]; $+0.850$ [+0.704, +0.996]; $+8.080$ [+7.829, +8.331]; $+0.283$ [+0.198, +0.367].
- D_{31}: $+4.592$ [+4.315, +4.870]; $+1.385$ [+1.293, +1.476]; $+10.163$ [+10.002, +10.323]; $+0.637$ [+0.450, +0.824]; $+10.936$ [+10.763, +11.109]; $+0.210$ [+0.143, +0.277].

Table 12: Checkpoint-level C8 contrasts (updates; equal-block descriptives). No confidence intervals are shown; these entries are descriptive and do not imply a training-seed population effect.

Length	Checkpoint	Mean–changing	Mean–preserving
8	Z_{61}/s0	-1.525	+4.998
8	Z_{61}/s1	-0.736	+2.658
8	Z_{61}/s3	-0.137	+1.539
8	Z_{61}/s4	-0.582	+8.807
8	$(\mathbb{Z}_2)^6$/s0	-0.602	+2.678
8	$(\mathbb{Z}_2)^6$/s1	-1.828	-0.168
8	$(\mathbb{Z}_2)^6$/s3	+0.178	+3.008
8	$(\mathbb{Z}_2)^6$/s4	-0.416	+5.936
8	D_{31}/s0	-0.953	+3.539
8	D_{31}/s1	-0.764	+5.279
8	D_{31}/s4	-0.604	+7.893
8	A_5/s0	-1.088	+7.803
8	A_5/s1	-0.348	+6.438
8	A_5/s2	-1.875	+7.477
8	A_5/s3	-0.395	+1.271
8	A_5/s4	-0.404	+4.826
6	Z_{61}/s0	-1.324	+2.656
6	Z_{61}/s1	-0.855	+2.049
6	Z_{61}/s3	-0.473	+1.146
6	Z_{61}/s4	-0.789	+5.871
6	$(\mathbb{Z}_2)^6$/s0	-0.621	+2.141
6	$(\mathbb{Z}_2)^6$/s1	-3.029	-0.869
6	$(\mathbb{Z}_2)^6$/s3	-0.504	+2.057
6	$(\mathbb{Z}_2)^6$/s4	-0.678	+3.168
6	D_{31}/s0	-0.992	+1.932
6	D_{31}/s1	-1.111	+1.469
6	D_{31}/s4	-0.646	+2.891
6	A_5/s0	-1.078	+2.408
6	A_5/s1	-0.539	+3.500
6	A_5/s2	-1.793	+5.375
6	A_5/s3	-0.668	+0.074
6	A_5/s4	-0.463	+3.221

- A_5: $+3.341$ [+3.071, +3.611]; $+0.871$ [+0.807, +0.936]; $+8.904$ [+8.783, +9.025]; $+0.282$ [+0.165, +0.399]; $+9.726$ [+9.617, +9.834]; $+0.037$ [+0.013, +0.061].

Length 6.

- Z_{61}: $+2.442$ [+2.180, +2.704]; $+1.164$ [+1.027, +1.300]; $+5.373$ [+5.241, +5.505]; $+0.562$ [+0.448, +0.675]; $+6.233$ [+6.136, +6.331]; $+0.109$ [+0.065, +0.153].
- $(\mathbb{Z}_2)^6$: $+4.210$ [+3.889, +4.531]; $+1.719$ [+1.589, +1.849]; $+5.834$ [+5.704, +5.964]; $+0.561$ [+0.423, +0.699]; $+7.042$ [+6.847, +7.237]; $+0.269$ [+0.187, +0.350].
- D_{31}: $+5.792$ [+5.379, +6.206]; $+2.056$ [+1.889, +2.222]; $+7.889$ [+7.741, +8.037]; $+0.441$ [+0.311, +0.572]; $+8.806$ [+8.637, +8.975]; $+0.171$ [+0.114, +0.229].
- A_5: $+3.718$ [+3.496, +3.940]; $+1.091$ [+1.010, +1.172]; $+6.634$ [+6.538, +6.729]; $+0.205$ [+0.118, +0.291]; $+7.542$ [+7.454, +7.629]; $+0.016$ [-0.017, +0.049].

C.4 Wrong-stop risk is accompanied by time and coverage

The tuples give *wrong-stop percentage of all pairs; capped D_8 lag; non-stop percentage* for the four plotted conditions in order: sham, preserving donor, fixed mean, changing donor. Lags include the insertion readout and cannot be below seven; non-events cap at 35. Rates and times here are descriptive. Registered wrong-stop contrasts are in Table 11; their family also includes non-stop and stopping-lag contrasts.

Length 8.

- Z_{61}: (0.098%; 7.229; 0.000%), (0.342%; 9.782; 0.000%), (0.195%; 14.293; 0.000%), (15.527%; 13.495; 0.000%).
- $(\mathbb{Z}_2)^6$: (0.000%; 7.228; 0.000%), (1.416%; 11.543; 0.000%), (0.098%; 14.633; 0.195%), (6.348%; 14.580; 0.000%).
- D_{31}: (0.130%; 7.176; 0.000%), (0.326%; 11.766; 0.000%), (0.391%; 17.329; 0.000%), (27.865%; 15.256; 0.000%).
- A_5: (0.469%; 7.228; 0.000%), (0.508%; 10.606; 0.000%), (0.469%; 16.173; 0.000%), (35.195%; 13.521; 0.000%).

Length 6.

- Z_{61}: (0.000%; 7.178; 0.000%), (0.049%; 9.616; 0.000%), (0.000%; 12.551; 0.000%), (3.516%; 13.104; 0.000%).
- $(\mathbb{Z}_2)^6$: (0.000%; 7.188; 0.000%), (0.977%; 11.247; 0.000%), (0.000%; 13.022; 0.195%), (2.637%; 13.840; 0.000%).
- D_{31}: (0.260%; 7.223; 0.000%), (0.391%; 13.023; 0.000%), (0.651%; 15.086; 0.000%), (2.539%; 15.843; 0.000%).
- A_5: (0.078%; 7.177; 0.000%), (0.000%; 10.916; 0.000%), (0.000%; 13.831; 0.000%), (1.758%; 14.573; 0.000%).

At length eight, changing-donor risk includes wrong predictions that persist long enough to trigger the rule, even if correctness returns later. A longer observation window cannot retroactively make an earlier wrong stop correct. Length-six wrong-stop rates are lower and stopping-time contrasts change. The eight-repeat rule is one evaluated rule, not a calibrated confidence threshold or an optimized stopping policy.

D Post-hoc analyses of the saved Experiment II logits

All analyses in this section are post hoc and descriptive; they were chosen after the registered analysis was complete and alter no registered criterion. Their definitions, rule grid and candidate labels were fixed in a written specification before any new outcome was computed. A single read-only

pass over all 16,384 archived pair files (member hashes checked) produced compact per-pair arrays, and one analysis script computed every value below; no resampling or randomness is involved. Two reproduction checks passed before any new number was used: the both-correct sets and their error shares (6,582 pairs, 53.36% versus 0.84% at length eight; 6,554 pairs, 56.29% versus 0.90% at length six) and the registered D_8 wrong-stop table (maximum absolute difference 0.0005 percentage points, i.e. rounding). Shares conditioned on both-correct pairs pool those pairs, so checkpoints are weighted by their both-correct counts (325–490 per checkpoint); other shares weight checkpoints equally.

Stopping rules. A repeat rule with window k and burn-in b stops at the first lag $t \leq 35$ with $t - k + 1 \geq b$ and k identical predictions ending at t; D_8 is $k = 8$, $b = 0$. A KL rule stops at the first $t \geq 1$ with $\mathrm{KL}(p_t \| p_{t-1}) < \tau$, where p_t is the softmax over all 128 output logits. The stop label is the prediction at the stopping lag; non-stops are capped at 35 as in the main analysis. No rule or threshold was selected, so all thirteen are reported. Non-stop rates are at most 13.5% in any task, condition and rule; per-task values are in the supplementary files.

Rule	Sham	Preserving	Fixed mean	Changing	Checkpoints
repeat k=4, b=0	0.4 / 3.2	2.0 / 4.1	12.8 / 10.6	72.8 / 5.5	16 / 16
repeat k=4, b=4	0.2 / 7.0	4.9 / 8.6	5.2 / 11.4	35.9 / 10.2	14 / 15
repeat k=4, b=8	0.2 / 11.0	2.2 / 11.9	0.9 / 12.2	2.6 / 12.8	6 / 13
repeat k=8, b=0	0.2 / 7.2	0.6 / 10.9	0.3 / 15.6	21.2 / 14.2	15 / 15
repeat k=8, b=4	0.2 / 11.0	0.6 / 13.9	0.2 / 15.7	1.0 / 16.3	6 / 9
repeat k=8, b=8	0.2 / 15.0	0.5 / 15.9	0.2 / 16.2	0.5 / 16.9	0 / 1
repeat k=12, b=0	0.2 / 11.2	0.5 / 16.3	0.2 / 19.6	0.6 / 20.3	5 / 6
repeat k=12, b=4	0.2 / 15.0	0.5 / 17.9	0.2 / 19.7	0.4 / 20.4	0 / 1
repeat k=12, b=8	0.2 / 19.0	0.4 / 20.0	0.2 / 20.2	0.4 / 20.9	0 / 1
KL $< 10^{-1}$	1.7 / 1.3	4.5 / 1.7	40.7 / 7.1	89.7 / 2.2	16 / 15
KL $< 10^{-2}$	1.0 / 1.6	3.7 / 2.2	14.6 / 9.6	82.0 / 3.2	16 / 15
KL $< 10^{-3}$	0.5 / 2.2	3.0 / 3.4	4.6 / 11.3	68.9 / 5.0	16 / 16
KL $< 10^{-4}$	0.4 / 3.6	2.1 / 5.6	1.8 / 12.8	51.2 / 7.7	16 / 16

Table 13: Label-free stopping rules at length eight: wrong-stop percentage of all pairs / mean stopping lag, averaged over the four tasks. The last column counts the sixteen checkpoints in which the changing donor stops wrongly more often than the preserving donor / than the fixed mean.

Rule	Sham	Preserving	Fixed mean	Changing	Checkpoints
repeat k=4, b=0	0.1 / 3.2	2.5 / 4.3	7.6 / 9.1	61.8 / 5.9	16 / 15
repeat k=4, b=4	0.1 / 7.0	3.8 / 8.7	1.8 / 9.7	8.7 / 10.2	10 / 12
repeat k=4, b=8	0.1 / 11.0	0.6 / 11.3	0.2 / 11.2	0.9 / 11.6	4 / 4
repeat k=8, b=0	0.1 / 7.2	0.4 / 11.2	0.2 / 13.6	2.6 / 14.3	11 / 11
repeat k=8, b=4	0.1 / 11.0	0.4 / 13.0	0.1 / 13.7	0.5 / 14.6	4 / 4
repeat k=8, b=8	0.1 / 15.0	0.3 / 15.3	0.1 / 15.2	0.4 / 15.6	2 / 2
repeat k=12, b=0	0.1 / 11.2	0.3 / 15.3	0.1 / 17.6	0.4 / 18.5	2 / 2
repeat k=12, b=4	0.1 / 15.0	0.3 / 17.0	0.1 / 17.7	0.4 / 18.6	2 / 2
repeat k=12, b=8	0.1 / 19.0	0.3 / 19.3	0.1 / 19.2	0.4 / 19.6	2 / 2
KL $< 10^{-1}$	1.1 / 1.3	4.0 / 1.6	30.7 / 6.2	86.8 / 2.3	16 / 15
KL $< 10^{-2}$	0.6 / 1.5	3.2 / 2.1	11.2 / 7.8	77.8 / 3.1	16 / 15
KL $< 10^{-3}$	0.3 / 1.9	2.9 / 2.9	3.4 / 8.9	66.2 / 4.3	16 / 16
KL $< 10^{-4}$	0.3 / 2.6	2.2 / 4.4	1.7 / 10.1	50.1 / 6.1	16 / 16

Table 14: Label-free stopping rules at length six: wrong-stop percentage of all pairs / mean stopping lag, averaged over the four tasks. The last column counts the sixteen checkpoints in which the changing donor stops wrongly more often than the preserving donor / than the fixed mean.

Which wrong label? Candidates were fixed from Eq. (1) before counting: $h_1 = (a_1 k)a_2 \cdots a_L$ (donor's first operand, recipient's remainder), $h_2 = a_1(k^{-1}a_2)a_3 \cdots a_L$, $h_1^- = (g a_1 k)a_2 \cdots a_L$, and the changing answer gP. For preserving errors gP is a control, because that donor never saw g. In $(\mathbb{Z}_2)^6$, $h_1 = h_2$ always. The uniform reference $1/(|G| - 1)$ is an assumption, not an estimate.

Certificate and retention. With the sham as reference, G_t is evaluated for five starts at every lag; a code check finds no case of $G_t = 1$ with a different argmax, as Lemma 1 requires. Over lags 0–8, G_t holds at 21–83% of updates for preserving donors and 0–73% for changing donors, across tasks

		Preserving: first error, lags 1–8				Changing: D_8 wrong stop			
Len.	Task	n	h_1	h_2	gP (control)	n	gP	h_1^-	Uniform
8	$\mathbb{Z}_{61}$	674	2.1	2.2	2.8	318	55.3	0.6	1.67
8	$(\mathbb{Z}_2)^6$	1125	5.7	5.7	1.3	130	54.6	3.1	1.59
8	D_{31}	741	1.5	0.8	2.4	428	97.7	0.2	1.64
8	A_5	972	4.3	1.6	1.3	901	98.0	0.1	1.69
6	$\mathbb{Z}_{61}$	748	2.8	2.4	1.6	72	66.7	1.4	1.67
6	$(\mathbb{Z}_2)^6$	1060	6.1	6.1	1.7	54	22.2	22.2	1.59
6	D_{31}	822	1.2	0.5	1.6	39	84.6	0.0	1.64
6	A_5	1059	6.7	2.0	1.6	45	97.8	0.0	1.69

Table 15: Share (%) of analysed wrong labels equal to each algebraic candidate. No analysed error was a non-group token. Changing-donor predictions at insertion equal gP in 83.9–89.9% of pairs.

and lengths. Stratifying by G_0 conditions on a post-intervention quantity and is descriptive. E_8 is the share of both-correct pairs with any error at lags 1–8; erring preserving continuations are wrong at 2.31 (length eight) and 2.52 (length six) of the eight lags on average.

Len.	Task	n	E_8 preserving	E_8 sham	G_0 holds	Error | G_0	Error | $\neg G_0$	Checkpoint E_8 range
8	$\mathbb{Z}_{61}$	1636	41.2	1.0	36.6	25.3	50.4	12.8–64.4 / 0.5–2.0
8	$(\mathbb{Z}_2)^6$	1682	66.9	0.9	79.1	62.6	83.2	51.8–81.7 / 0.0–1.9
8	D_{31}	1257	58.9	0.3	43.3	48.7	66.8	39.1–75.5 / 0.0–0.5
8	A_5	2007	48.4	0.9	35.5	34.6	56.0	32.6–90.7 / 0.0–3.4
6	$\mathbb{Z}_{61}$	1697	44.1	0.8	40.7	30.5	53.4	21.1–65.8 / 0.0–2.1
6	$(\mathbb{Z}_2)^6$	1649	64.3	0.5	76.5	60.9	75.2	51.7–83.4 / 0.0–0.9
6	D_{31}	1150	71.5	1.0	38.9	70.9	71.8	51.8–81.2 / 0.0–2.4
6	A_5	2058	51.5	1.2	30.4	40.0	56.5	29.3–90.8 / 0.0–6.1

Table 16: Both-correct pairs: error shares, certificate at insertion, and the per-checkpoint range of E_8 (preserving / sham). Percentages.

E COMPLETE INDEPENDENT-POOL REPLICATION RESULTS

Starting state	Construction	Evaluation-donor dependence
Computed donor	d	Full state
Native reset	h_0	None
Isotropic	$N_{\|d\|}(z_0)$	Norm only
Mean + noise, paired scale	$N_{\|d\|}(\mu + \sigma z)$	Norm only; primary arm
Mean only, paired scale	$N_{\|d\|}(\mu)$	Norm only
Mean + noise, pool scale	$N_{r_p}(\mu + \sigma z)$	None
Mean only, pool scale	$N_{r_p}(\mu)$	None; fixed within block

Table 17: **Experiment I starting states. A fixed mean removes both donor coordinates and donor scale.** z_0, z are standard isotropic Gaussian draws; noisy conditions share their draw across the two scales. All conditions continue under the recipient prompt. Experiment II retains the fixed mean and native reset, uses pool-scaled isotropic noise, and adds sham and answer-preserving donors.

The new study has 32 independent construction/evaluation blocks per task, with evaluation contexts synchronized across a task's checkpoints. Every condition has 512 evaluation pairs per checkpoint. The tables below preserve checkpoint-specific results and both registered inference families. All new intervals use the repeated blocks, including newly fitted means, variances and construction-pool norms. This is distinct from the discovery archives described in the subsequent appendices.

The primary mean-plus-noise versus donor analysis uses 50,000 bootstrap draws, seed 20260922, and a common critical value of 2.9486822652 across twenty effects. The secondary family uses 20,000 draws, seed 20260923, and a common critical value of 3.8434006553 across 440 effects. Cells with zero observed block variance have no inferential interval. The family labels and zero-

Task	Start	A_{27}	S_{27}	C_8	W_8	Hit (%)	Suffix (%)	Run (%)
Z61	Donor	7.69	8.04	15.03	0.19	99.90	99.90	99.90
Z61	Native	6.88	8.49	15.49	1.09	99.90	99.90	99.90
Z61	Isotropic, paired	7.98	10.23	17.23	1.52	99.90	99.80	99.76
Z61	Mean+noise, paired	5.78	7.41	14.40	1.07	99.90	99.85	99.90
Z61	Mean only, paired	5.77	7.40	14.35	1.05	99.90	99.80	99.90
Z61	Mean+noise, fixed	5.76	7.41	14.40	1.08	99.90	99.85	99.90
Z61	Mean only, fixed	5.75	7.37	14.32	1.04	99.90	99.71	99.90
Z26	Donor	7.97	8.46	15.48	0.35	98.73	98.39	98.44
Z26	Native	8.08	9.79	16.79	1.18	99.80	99.80	99.61
Z26	Isotropic, paired	8.03	9.77	16.78	1.20	99.71	99.61	99.61
Z26	Mean+noise, paired	6.12	7.87	14.89	1.05	99.80	98.58	98.14
Z26	Mean only, paired	6.22	7.81	14.81	1.04	99.85	99.71	99.41
Z26	Mean+noise, fixed	6.13	7.86	14.88	1.05	99.71	98.58	98.14
Z26	Mean only, fixed	6.21	7.77	14.77	1.03	99.90	99.80	99.51

Table 18: Absolute outcomes, equal-checkpoint task means: Z61, Z26. Latencies are capped; event frequencies distinguish non-events. W_8 is the eight-update post-hit error count with non-hits coded as eight. These descriptive means are not additional hypothesis tests.

Task	Start	A_{27}	S_{27}	C_8	W_8	Hit (%)	Suffix (%)	Run (%)
D31	Donor	10.73	11.27	18.28	0.29	99.80	99.09	99.09
D31	Native	10.50	11.86	18.84	0.88	100.00	99.09	99.22
D31	Isotropic, paired	11.05	12.64	19.63	1.03	99.80	99.09	99.22
D31	Mean+noise, paired	9.33	10.47	17.45	0.70	99.80	99.09	99.22
D31	Mean only, paired	9.35	10.49	17.47	0.67	99.87	99.09	99.22
D31	Mean+noise, fixed	9.34	10.48	17.46	0.70	99.80	99.09	99.22
D31	Mean only, fixed	9.35	10.50	17.49	0.68	99.87	99.09	99.22
A5	Donor	9.92	10.07	17.08	0.11	99.57	99.45	99.45
A5	Native	10.76	11.83	18.83	0.75	99.84	99.53	99.53
A5	Isotropic, paired	11.00	12.48	19.49	0.98	99.92	99.49	99.45
A5	Mean+noise, paired	8.54	9.39	16.40	0.57	99.88	99.53	99.45
A5	Mean only, paired	8.52	9.24	16.24	0.47	99.92	99.53	99.53
A5	Mean+noise, fixed	8.53	9.39	16.39	0.58	99.88	99.53	99.45
A5	Mean only, fixed	8.50	9.21	16.21	0.46	99.92	99.53	99.53

Table 19: Absolute outcomes, equal-checkpoint task means: D31, A5. Latencies are capped; event frequencies distinguish non-events. W_8 is the eight-update post-hit error count with non-hits coded as eight. These descriptive means are not additional hypothesis tests.

variance policy are included in the machine-readable tables, not inferred from rounded printed numbers.

The timing decomposition used in the discovery and confirmatory tables is the arithmetic identity

$$\Delta S_H = \Delta A_H + \Delta B_H, \qquad B_H = S_H - A_H. \tag{2}$$

It does not identify a causal mediator. The fixed-window observations test the unequal-opportunity explanation directly.

Selection in a both-hit description. For each checkpoint, the archive also restricts the paired comparison to pairs where both donor and mean-plus-noise produce a hit through 27. This sensitivity calculation preserves paired outcomes and reports the selected pair/context denominators. Because hitting is affected by the intervention, the result is descriptive rather than a controlled direct effect. The main analysis instead uses every pair and explicitly reports its non-hit component.

Implementation and data audit. The local and GPU-host CPU regression suites each passed 21 tests. All sixteen checkpoint-specific preflights passed before the main queue. The prospective-window implementation includes boundary cases with first hit at lag 27, no hit, intermittent correctness and runs ending at lag 35. A separate postprocessor reconstructs the saved endpoints from argmaxes of the complete logit vectors; another reconstructs block matrices and reruns both simultaneous-inference procedures. Neither imports the measurement or analysis module.

Independent numerical reconstruction. All 8,940 transferred files passed SHA-256 comparison. The independent array-level audit reconstructed 2,064,384 logit argmaxes, all seven-condition

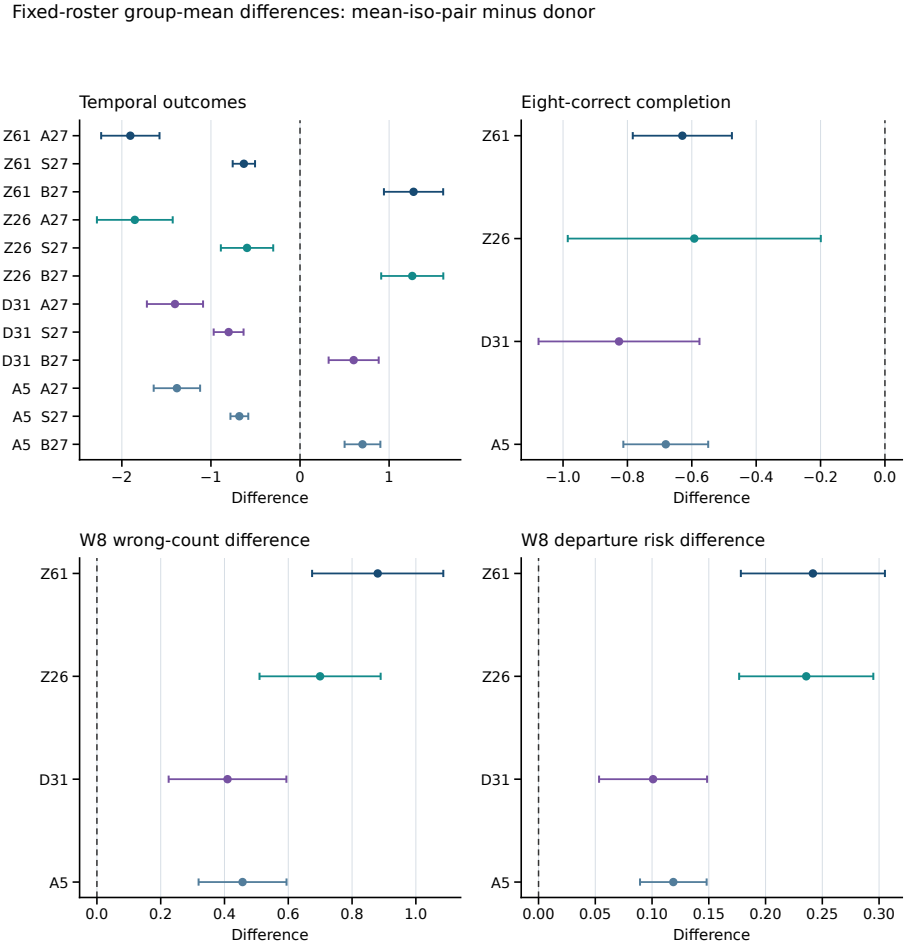


Figure 8: **Complete timing comparison for the task means.** Mean-plus-noise at donor scale minus donor. S_{27} has primary-family intervals; arrival, gap, eight-correct completion, and fixed-window counts/risks have secondary-family intervals. Positive post-hit counts coexist with negative completion delays. These are outcomes of the full interventions, not identified effects on a common hit state.

Task	Checkpoint	Estimate	Simultaneous 95% CI	Status
Z61	seed 0	-1.47	[-1.70, -1.23]	upper bound < 0
Z61	seed 1	-0.22	[-0.45, 0.02]	not established
Z61	seed 3	0.03	[-0.33, 0.38]	not established
Z61	seed 4	-0.86	[-1.24, -0.48]	upper bound < 0
Z61	roster mean	-0.63	[-0.76, -0.50]	upper bound < 0
Z26	seed 0	-0.57	[-0.92, -0.22]	upper bound < 0
Z26	seed 1	-2.47	[-3.18, -1.77]	upper bound < 0
Z26	seed 3	0.13	[-0.28, 0.54]	not established
Z26	seed 4	0.54	[-0.12, 1.19]	not established
Z26	roster mean	-0.59	[-0.89, -0.30]	upper bound < 0
D31	seed 0	-0.98	[-1.35, -0.62]	upper bound < 0
D31	seed 1	-0.96	[-1.21, -0.72]	upper bound < 0
D31	seed 4	-0.46	[-0.60, -0.32]	upper bound < 0
D31	roster mean	-0.80	[-0.97, -0.63]	upper bound < 0
A5	seed 0	-0.60	[-0.86, -0.34]	upper bound < 0
A5	seed 1	-0.43	[-0.55, -0.31]	upper bound < 0
A5	seed 2	-1.74	[-2.14, -1.33]	upper bound < 0
A5	seed 3	-0.52	[-0.71, -0.33]	upper bound < 0
A5	seed 4	-0.12	[-0.30, 0.06]	not established
A5	roster mean	-0.68	[-0.78, -0.58]	upper bound < 0

Table 20: Primary mean-iso$_{\mathrm{pair}}$ minus donor contrast in S_{27} (updates). Intervals are simultaneous centered block max-t 95% intervals over the 20-cell primary family. A zero-observed-SE cell is marked unavailable; an interval spanning zero is not evidence of equivalence. Negative values favor mean-iso$_{\mathrm{pair}}$.

Outcome	Z61	Z26	D31	A5
A27	-1.90 [-2.23, -1.58]	-1.85 [-2.28, -1.43]	-1.40 [-1.72, -1.09]	-1.38 [-1.64, -1.12]
S27	-0.63 [-0.76, -0.50]	-0.59 [-0.89, -0.30]	-0.80 [-0.97, -0.63]	-0.68 [-0.78, -0.58]
B27	1.27 [0.94, 1.61]	1.26 [0.91, 1.61]	0.60 [0.32, 0.88]	0.70 [0.50, 0.90]
C8	-0.63 [-0.78, -0.48]	-0.59 [-0.99, -0.20]	-0.83 [-1.08, -0.58]	-0.68 [-0.81, -0.55]
W_8 wrong count	0.88 [0.67, 1.09]	0.70 [0.51, 0.89]	0.41 [0.22, 0.59]	0.46 [0.32, 0.59]
W_8 any departure	0.24 [0.18, 0.31]	0.24 [0.18, 0.29]	0.10 [0.05, 0.15]	0.12 [0.09, 0.15]

Table 21: Mean-iso$_{\mathrm{pair}}$ minus donor fixed-roster group-mean contrasts. Entries are estimate [simultaneous 95% CI]. A_{27}, S_{27}, B_{27} and C_8 are update counts; W_8 reports error-count difference and any-departure risk difference. S_{27} uses the 20-cell primary family interval; all other rows use the 440-cell secondary family. W_8 counts errors in the eight updates after first hit by lag 27; pairs with no hit by 27 are separately flagged and assigned eight errors in the unconditional composite. Intervals crossing zero do not establish equivalence.

endpoints for 8,192 pairs, and 512 checkpoint/block summaries; the maximum discrepancy in reconstructed block metrics was exactly zero. Both registered bootstrap critical values and all 460 inferential cells were independently reproduced from these block values, including available intervals and unavailable zero-variance cells. All sixteen same-shape preflight logit errors were zero; the largest norm-rescaling relative error was 1.44×10^{-8} against the frozen 2×10^{-6} tolerance. The fixed mean-only state's hash was identical across the sixteen evaluation pairs in every block, confirming the within-block constant start in the saved execution.

Exact archived eligibility and stage rules. Each checkpoint's prior calibration used 512 independent contexts. Its capability gate required accuracy at update 27 of at least $\max\{0.80, \mathrm{Acc}_{27}(s0) - 0.05\}$, using the archived task-specific reference accuracy. The resulting thresholds are 0.95 for Z61, Z26 and A5, and 0.941935484 for D31. The calibrated t_{80} is the earliest update whose accuracy and the next two updates' accuracies are all at least 0.80. Prior eligibility also required native parity and an available $t_{80} \leq 23$. Frozen stages are Z61: 11/11/6/10 for s0/1/3/4; Z26: 10/10/10/15 for s0/1/3/4; D31: 15/12/12 for s0/1/4; and A5: 13/12/13/12/13 for s0–s4. These rules selected the archived roster before this study. The new singleton numerical preflight is a separate execu-

Contrast	Z61	Z26	D31	A5
Isotropic Gaussian vs donor	2.19 [1.99, 2.38]	1.31 [1.01, 1.61]	1.37 [1.12, 1.63]	2.41 [2.28, 2.54]
Mean + noise, paired vs donor	-0.63 [-0.76, -0.50]	-0.59 [-0.89, -0.30]	-0.80 [-0.97, -0.63]	-0.68 [-0.78, -0.58]
Mean only, paired vs donor	-0.65 [-0.86, -0.43]	-0.66 [-1.01, -0.31]	-0.78 [-1.03, -0.54]	-0.84 [-0.98, -0.69]
Mean + noise, fixed vs donor	-0.63 [-0.79, -0.47]	-0.61 [-0.99, -0.22]	-0.79 [-1.00, -0.59]	-0.68 [-0.80, -0.56]
Mean only, fixed vs donor	-0.68 [-0.88, -0.48]	-0.69 [-1.04, -0.35]	-0.77 [-1.02, -0.51]	-0.87 [-1.01, -0.72]
Mean addition (paired scale)	-2.82 [-3.00, -2.63]	-1.90 [-2.18, -1.62]	-2.17 [-2.34, -2.01]	-3.09 [-3.23, -2.94]
Noise addition (paired scale)	0.01 [-0.11, 0.14]	0.06 [-0.17, 0.30]	-0.02 [-0.15, 0.11]	0.16 [0.06, 0.26]
Noise addition (fixed scale)	0.05 [-0.08, 0.18]	0.09 [-0.15, 0.32]	-0.03 [-0.16, 0.10]	0.18 [0.07, 0.29]
Scale effect (mean + noise)	-0.00 [-0.06, 0.05]	-0.01 [-0.13, 0.10]	0.01 [-0.03, 0.05]	-0.00 [-0.04, 0.03]
Scale effect (mean only)	-0.03 [-0.11, 0.04]	-0.04 [-0.09, 0.02]	0.02 [-0.02, 0.05]	-0.03 [-0.06, 0.00]
Noise $\times$ scale interaction	-0.03 [-0.10, 0.04]	-0.02 [-0.16, 0.11]	0.01 [-0.05, 0.06]	-0.03 [-0.07, 0.02]

Table 22: Equal-weight fixed-roster task-mean contrasts in S_{27} (updates), shown as estimate [simultaneous 95% CI]. The primary mean-plus-isotropic paired-scale contrast uses the 20-cell family; construction and scale contrasts use the 440-cell secondary family. Noise additions compare mean-plus-noise with the corresponding deterministic mean; mean addition compares mean-plus-noise with the isotropic Gaussian. Negative values favor the first construction in the contrast.

tion check, not a new capability screen. The frozen `CHECKPOINTS.json` retains the exact prior curves, gate decisions and calibration receipts.

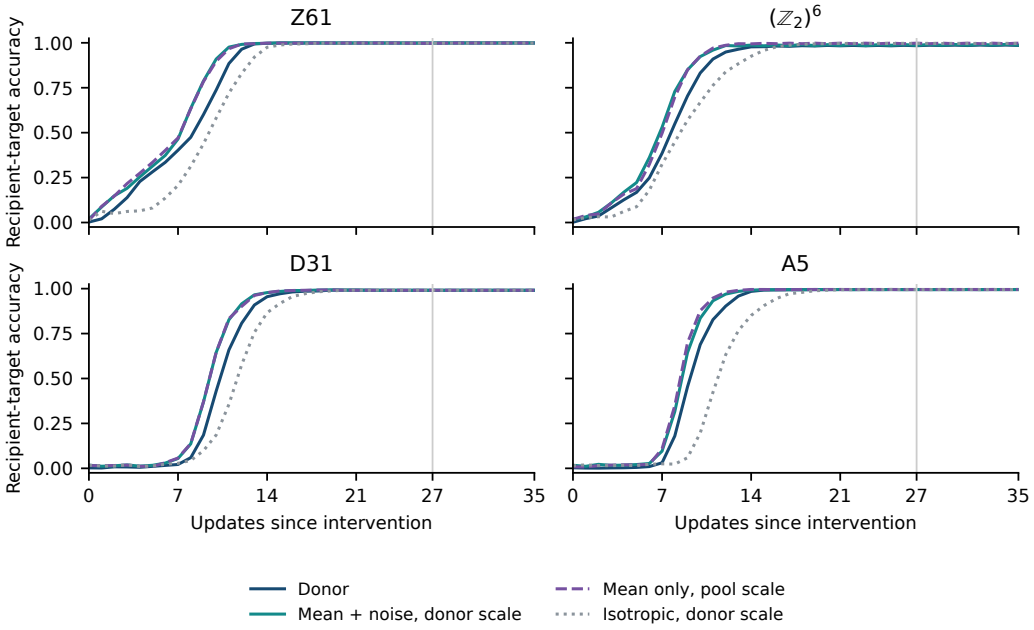


Figure 9: **Every-lag recipient accuracy in the new replication.** Curves are descriptive equal-checkpoint task means over all evaluation pairs. The vertical line is the primary horizon 27; observations continue to 35. Unconditional accuracy curves average across individuals and are distinct from the paired post-hit error counts. No additional interval or pointwise significance test is inferred from these curves.

E.1 Registered timing contrasts for the original noisy arm

Table 23 gives the task-mean contrasts of the registered mean-plus-noise arm against the real donor. Every interval for arrival, correct suffix and completion lies below zero, and every post-hit interval lies above zero: the noisy mean reaches and completes the answer earlier while making more errors after the first hit.

Task	ΔA_{27}	ΔS_{27}	ΔW_8	ΔC_8
Z61	-1.90 [-2.23, -1.58]	-0.63 [-0.76, -0.50]	+0.88 [+0.67, +1.09]	-0.63 [-0.78, -0.48]
$(\mathbb{Z}_2)^6$	-1.85 [-2.28, -1.43]	-0.59 [-0.89, -0.30]	+0.70 [+0.51, +0.89]	-0.59 [-0.99, -0.20]
D31	-1.40 [-1.72, -1.09]	-0.80 [-0.97, -0.63]	+0.41 [+0.22, +0.59]	-0.83 [-1.08, -0.58]
A5	-1.38 [-1.64, -1.12]	-0.68 [-0.78, -0.58]	+0.46 [+0.32, +0.59]	-0.68 [-0.81, -0.55]

Table 23: **Earlier completion, more post-hit errors.** Mean-plus-noise at donor scale minus donor; equal-checkpoint task means, in updates or incorrect readouts (W_8). Each cell shows estimate and simultaneous 95% interval. S_{27} uses the 20-cell primary family; other columns use the 440-cell secondary family. W_8 follows every hit through 27 for exactly eight further updates; non-hits receive eight errors and are reported separately.

F Proof and audits of the finite-response certificate

Details and sharpness of Lemma 1. *Proof of the lemma.* $m_t^v \geq m_t + d_{t,y} - \max_{j \neq y} d_{t,j} \geq m_t - \mathrm{osc}(d_t)$; wrong labels need $m_t^v \leq 0$. Put $\Psi_t = \ell_c \circ F_c^t$ for reference start $h^\star$ and alternative $h^\star + v$. If Ψ_t is continuously differentiable on the joining segment, $d_t = \Psi_t(h^\star + v) - \Psi_t(h^\star) = M_t v$ exactly, with $M_t = \int_0^1 D\Psi_t(h^\star + sv)\, ds$. Let a_t be the reference winner and $G_t = \mathbf{1}\{\mathrm{osc}(d_t) < m_t\}$ the certificate. For every rival j, the alternative gap from a_t is at least $m_t - (d_{t,j} - d_{t,a_t}) \geq m_t - \mathrm{osc}(d_t) > 0$. A correct reference and agreement imply correctness; eight consecutive such updates bound C_8. For sharpness, reference logits $(m, 0)$ and displacement $(0, m + \epsilon)$ reverse the winner for every $\epsilon > 0$, so no threshold $\mathrm{osc}(d) < \kappa m$ with $\kappa > 1$ uniformly suffices. The oscillation satisfies $\mathrm{osc}(d) = 2 \min_a \|d - a\mathbf{1}\|_\infty$.

Summing the error bound over the eight updates after an alternative's first hit bounds W_8. Uncertified behavior comes from a wrong reference or a displacement not dominated by its margin; a failed certificate is not itself an error. Retaining signed rival gaps gives the necessary and sufficient criterion for a unique target winner:

$$\Psi_{t,y}(h^\star) - \Psi_{t,j}(h^\star) > d_{t,j} - d_{t,y} \quad \text{for every } j \neq y. \tag{3}$$

Ties follow the readout rule. First hit requires this once; a correct run requires it repeatedly. Neither current correctness nor state norm alone guarantees either time ordering. Saved logits identify $M_t v$, not the full matrix or a local state Jacobian.

Historical four-checkpoint audit. The earlier numerical check uses the original four reference checkpoints and 1,024 pair records, each with donor, native restart and norm-matched random conditions plus the uninterrupted recipient control. It is distinct from the sixteen-checkpoint synthetic-state confirmation and is post hoc. We recompute full-vocabulary argmaxes from saved logits, using token IDs rather than group labels. A certificate requires a strictly positive residual margin and, for correctness, a correct reference. An absent certified suffix remains absent rather than being encoded as an event at the horizon.

Across 3,072 condition–pair records and 86,016 updates, the oscillation guard holds at 48,197 updates, compared with 46,989 for the original $2\|d_t\|_\infty$ guard. There are zero agreement, correctness or post-hit error-bound violations. This is a mathematical consistency check, not independent evidence of a predicted mechanism. The guard fails at 5,019 updates where the alternative is nevertheless correct: the sufficient certificate is not an equivalence. Common-logit-shift tests, absent-

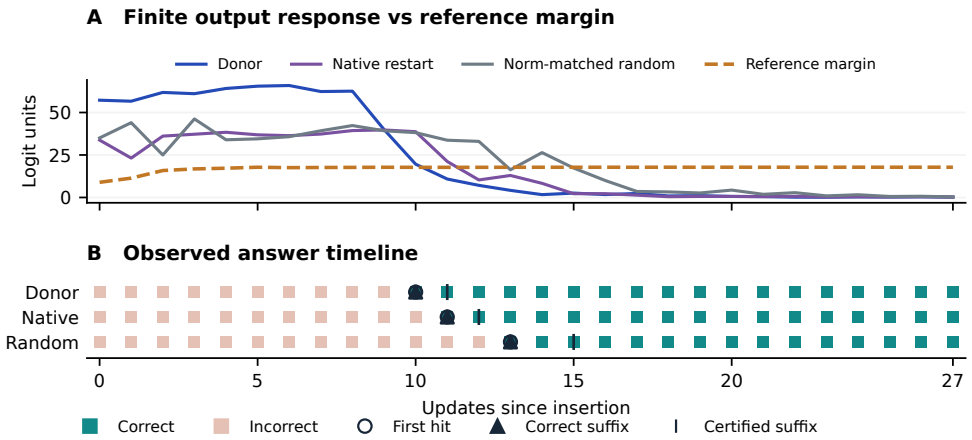


Figure 10: **Correctness can precede a margin certificate.** One fixed reference-checkpoint example (A5, first recorded context/action; no outcome selection). The upper panel compares the observed finite response $\mathrm{osc}(d_t)$ with the reference margin. The lower strips show actual correct/incorrect labels and first arrival, correct-suffix onset and the certified suffix. All curves use the same recipient continuation. This is a retrospective illustration from the reference archive, separate from the sixteen-checkpoint synthetic-state experiment.

Reference checkpoint	Pairs	Mean capped S_{27}	Events / censored
$\mathbb{Z}_{61}$	256	0.172	256 / 0
$(\mathbb{Z}_2)^6$	256	0.234	256 / 0
D_{31}	256	0.609	252 / 4
A_5	256	0.242	256 / 0

Table 24: **Uninterrupted recipient baseline in the old four-checkpoint archive.** Descriptive capped S_{27} in updates, reconstructed from 1,024 saved full-logit trajectories with horizon 27. Event and censor flags match the saved records. This historical reference is separate from the independent 16-checkpoint study.

certificate tests, wrong-reference tests and synthetic-array implication checks precede the archive audit.

Task	Start	Correct suffix events	Certificate events	Mean certificate slack
Z61	donor	256/256	256/256	1.121
Z61	native	256/256	256/256	1.383
Z61	random	256/256	256/256	1.160
Z26	donor	256/256	256/256	0.492
Z26	native	256/256	256/256	0.734
Z26	random	256/256	256/256	0.699
D31	donor	252/256	252/256	1.925
D31	native	252/256	252/256	2.484
D31	random	252/256	252/256	2.298
A5	donor	256/256	256/256	1.227
A5	native	256/256	256/256	1.602
A5	random	256/256	256/256	1.562

Table 25: Certificate availability and conservatism in the reference archive. Slack is $\tau_H - S_H$ conditional on a certificate being available; this conditioning is descriptive. No new inferential family or predictive-performance claim is attached.

The segment integral identifies $M_t v$ from output differences, not the operator M_t. Its dependence on v matters: two inserted states need not have a common linear response map. The guard does

not require non-normal growth, state contraction, or shrinking $\|v\|$. It is also not an early forecast: calculating the realized suffix crossing uses future logits. A future-envelope bound estimated before those logits, or a held-out forecast with initial-margin/arrival baselines, would be a separate predictive test. The reference being wrong prevents a correct-answer guarantee even when agreement is exact.

The earlier uncentered crossing check did not uniquely track the post-arrival component (six of sixteen archived contrasts aligned with that component). We therefore do not infer that the certified crossing explains stabilization specifically rather than the complete output sequence. The new sixteen-checkpoint study establishes its temporal contrasts through equal-follow-up observations, independently of this retrospective bound.

F.1 Exact routes of the finite response

Setting. Fix the recipient's prompt, $F_c = F(\cdot, e_c)$, and write $h = (u, z)$, with u the upstream positions $0, \dots, L$ and z the answer position $L{+}1$; z updates to $f_c(u, z)$, and the exit computes the logits $\ell_c(u, z)$ from both. Let $h_t = (u_t, z_t)$ be the untouched run and $\tilde{h}_t = (\tilde{u}_t, \tilde{z}_t)$ the transplant ($h^{\mathrm{S}}(t)$ and $h^{\mathrm{P}}(t)$ in Appendix G.11), both under F_c from insertion ($t = 0$), with $\delta h_t = (\delta u_t, \delta z_t) = \tilde{h}_t - h_t$ and $d_t = \ell_c(\tilde{h}_t) - \ell_c(h_t)$ as in Lemma 1. Assume f_c and ℓ_c are continuously differentiable (in real arithmetic, the ReLU-squared MLPs, RMSNorm with $\epsilon > 0$ and softmax attention of Appendix H are), and average the update's partial derivatives along the segment $h_t + \theta\,\delta h_t$, $\theta \in [0, 1]$, and the readout's along the two legs $h_t \to (u_t, \tilde{z}_t) \to \tilde{h}_t$: $\bar{f}_t^{\,z} = \int_0^1 \partial_z f_c(h_t + \theta\,\delta h_t)\,d\theta$ and $\bar{f}_t^{\,u}$ likewise, $\bar{\ell}_t^{\,z} = \int_0^1 \partial_z \ell_c(u_t, z_t + \theta\,\delta z_t)\,d\theta$ and $\bar{\ell}_t^{\,u} = \int_0^1 \partial_u \ell_c(u_t + \theta\,\delta u_t, \tilde{z}_t)\,d\theta$. Let $\Pi_{t,s} = \bar{f}_{t-1}^{\,z} \cdots \bar{f}_s^{\,z}$ for $t > s$ and $\Pi_{s,s} = I$.

Proposition 1 (Exact finite-response routes). *(i) At every lag $t \geq 0$,*

$$d_t = \underbrace{\bar{\ell}_t^{\,z}\Big(\Pi_{t,0}\,\delta z_0 + \sum_{s=0}^{t-1} \Pi_{t,s+1}\,\bar{f}_s^{\,u}\,\delta u_s\Big)}_{\textit{answer route}} + \underbrace{\bar{\ell}_t^{\,u}\,\delta u_t}_{\textit{upstream route}} \quad . \tag{4}$$

(ii) For a label y, let $m_{j,t} = \ell_{c,y}(h_t) - \ell_{c,j}(h_t)$, of either sign. The transplant reads y as the unique winner at lag t if and only if $d_{t,j} - d_{t,y} < m_{j,t}$ for every rival $j \neq y$, with d_t given by (4).

Proof. (i) By the fundamental theorem of calculus and the chain rule along $\theta \mapsto h_t + \theta\,\delta h_t$, $\delta z_{t+1} = f_c(\tilde{h}_t) - f_c(h_t) = \int_0^1 (\partial_z f_c\,\delta z_t + \partial_u f_c\,\delta u_t)\,d\theta = \bar{f}_t^{\,z}\,\delta z_t + \bar{f}_t^{\,u}\,\delta u_t$, and along the two legs $d_t = [\ell_c(u_t, \tilde{z}_t) - \ell_c(h_t)] + [\ell_c(\tilde{h}_t) - \ell_c(u_t, \tilde{z}_t)] = \bar{\ell}_t^{\,z}\,\delta z_t + \bar{\ell}_t^{\,u}\,\delta u_t$. Induction on t gives $\delta z_t = \Pi_{t,0}\,\delta z_0 + \sum_{s<t} \Pi_{t,s+1}\bar{f}_s^{\,u}\,\delta u_s$: it holds at $t = 0$, and multiplying it by $\bar{f}_t^{\,z}$ and adding $\bar{f}_t^{\,u}\,\delta u_t = \Pi_{t+1,t+1}\bar{f}_t^{\,u}\,\delta u_t$ gives it at $t{+}1$, because $\bar{f}_t^{\,z}\Pi_{t,s} = \Pi_{t+1,s}$. Substituting it into d_t proves (4). (ii) The transplant's lead of y over j is $m_{j,t} - (d_{t,j} - d_{t,y})$, and y is the unique winner exactly when every such lead is positive; this is Eq. (3) with $h^\star = h_0$. □

What the identity is. It is the standard mean-value form of variation of constants: exact for finite differences, with no linearisation. It places no condition on the upstream dynamics: causal attention makes the update of u independent of z (Section 4), but feedback from z would leave the identity intact. Its coefficients are averages along paths between the two realised runs' states, so it accounts for a change once both runs exist and forecasts nothing. The two terms are an accounting, not separately identified causal effects: intervening on one route changes the transplant run and with it the coefficients. In floating point, each update adds a rounding residual that the products $\Pi_{t,s}$ carry forward. Unlike $d_t = M_t v$ above, which treats $\ell_c \circ F_c^t$ as one map, it averages each update, which exposes the routes.

The two routes in the experiments. Operand-only transplants have $\delta z_0 = 0$: the answer route is what the answer position has stored of earlier upstream differences, and the upstream route is what the exit reads from the current one. The answer route is $\ell_c(u_t, \tilde{z}_t) - \ell_c(h_t)$, the logit change behind the *ansact* cross readout $R(u_S, a_c)$ (Appendix G.2, where S is the untouched run, c the transplant and a the answer position; only its label is saved), and the *upact* change $\ell_c(\tilde{u}_t, z_t) - \ell_c(h_t)$ differs from the upstream route by the interaction $\ell_c(\tilde{h}_t) - \ell_c(\tilde{u}_t, z_t) - \ell_c(u_t, \tilde{z}_t) + \ell_c(h_t)$. Taking the legs

in the other order moves the interaction into the answer route; the split depends on this choice, while d_t and (ii) do not. At the first error, ansact reproduces the transplant's wrong label and upact reads the target in the pair of Figure 4b; in that of c the two swap (Section 4.3). The effective map moves in b, the boundary in c.

Relation to Lemma 1 and Proposition 2. The lemma's test $\mathrm{osc}(d_t) < m_t$ implies the condition of Proposition 1(ii), since $d_{t,j} - d_{t,y} \le \mathrm{osc}(d_t)$ and $m_t = \min_{j \neq y} m_{j,t}$ when the untouched run reads y. For each fixed t, as $v = \delta h_0 \to 0$, $\bar{f}_t^z \to J_t[L{+}1, L{+}1]$ and $\bar{f}_t^u \to (J_t[L{+}1, j])_{j \le L}$, blocks of the Jacobians of Proposition 2. Proposition 2 gives sufficient conditions for a causal recurrence to act as a delay line; Proposition 1 needs none of these conditions and splits the realised logit change d_t between the two routes.

Continuation sets. With $q_c(h) = \arg\max_j \ell_{c,j}(h)$ and $\mathcal{R}_y = \{h : q_c(h) = y\}$, the states that read y now and after each of the next H updates form $\mathcal{R}_y^H = \bigcap_{0 \le t \le H} F_c^{-t}(\mathcal{R}_y) \subseteq \mathcal{R}_y$. Up to ties, which follow the readout rule, (ii) at $t = 0, \dots, H$ shows that the transplant starts in $\mathcal{R}_y^H$ if and only if $d_{t,j} - d_{t,y} < m_{j,t}$ for all $t \le H$ and $j \neq y$. With y the target, a both-correct transplant that errs by lag H starts in $\mathcal{R}_y \setminus \mathcal{R}_y^H$: equal current answers establish membership in $\mathcal{R}_y$, not in $\mathcal{R}_y^H$. The slice $\{z : q_c(u, z) = y\}$ depends on u; for $u = \tilde{u}_t$, its trace on the line of Figure 4c is the black boundary that moves there. Lemma 1 certifies membership in $\mathcal{R}_y$ at one lag; certifying $\mathcal{R}_y^H$ at insertion needs advance bounds on $d_{t,j} - d_{t,y}$ and $m_{j,t}$ for $t = 1, \dots, H$. $\mathcal{R}_y^H$ is a nonlinear, finite-horizon analogue of the output admissible sets of linear control (Gilbert & Tan, 1991). It can be a strict subset of $\mathcal{R}_y$ even when no update enlarges a distance (the example in Appendix G.11), so late errors from correct states do not by themselves indicate chaos. Nor does any rule on the labels read so far give the next label under a fixed prompt: where the untouched run reads the target throughout, a both-correct transplant has read the same labels as it up to its first error, at which the two differ. Exits that count repeated labels (Section 6) are therefore calibrated on data, not certified by the labels.

G MECHANISM EXPERIMENTS: METHODS, PREDICTIONS AND COMPLETE RESULTS

Every analysis in this appendix is exploratory [X]. We designed them after the outcomes of Experiment II were known and ran them on that experiment's pairs. We score the 24 predictions whose rules were written into analysis scripts before those scripts' first run (P1–P3, Pa–Pd, Pb′, E1–E5, EM1–EM5, PK1–PK3, X1a–X1c) with their rules as written, including every miss and every clause that holds by construction (Appendix G.3). Two later scripts (the delay line with the power iteration, and the timed clamps with the frozen answer map) also stated predictions before their first run but were analysed after that tally had been compiled. The main text reports them, and Table 28 scores their predictions by their rules as written, including a miss, outside that tally. Four further scripts stated predictions whose deciding statistics or options were not fixed before running; we report them as exploratory. Everything else, including every re-analysis, control and summary written after a run, is post hoc. LLM coding agents of one model family (Claude) implemented and cross-checked all analyses, including the separate re-analyses cited below; no human and no other model has re-implemented them. The order of predictions and outputs rests on file times and session records kept by the same pipeline, not on external time stamps.

The appendix follows the main text. Appendices G.1–G.7 give the pairs, the interventions, the verdicts and the complete tables behind Section 4. Appendix G.9 reports two controls on insertion time and displacement direction. Appendix G.10 gives methods and complete results for the tests of a transient-chaos reading in Section 5. Appendix G.12 lists the accounts we tested and did not support, and Appendix G.13 the open discrepancies between specifications and code.

G.1 PAIRS, READOUT AND OUTCOME

We rebuild the answer-preserving pairs of Experiment II with the frozen experiment code, imported read-only. We use the first four evaluation blocks; each block has eight contexts and each context two edits, which gives $4 \times 8 \times 2 = 64$ pairs per checkpoint and length for all 16 checkpoints at $L = 8$ and $L = 6$. Positions are 0 (BOS), $1, \ldots, L$ (operands) and $L+1$ (answer). The edit changes operand positions 1 and 2, which we call the *seam*. At the checkpoint's insertion update t_{80}, h_S is the recipient's own state (the sham), h_P is the answer-preserving donor state, and $v = h_P - h_S$. Every start is rolled out for 35 updates (24 in the effective-map and cross-readout runs) under the recipient prompt. At each lag the readout applies the exit blocks to all positions and then reads the last position, so the readout is a function of the whole state, not of the answer position alone.

All interventions of Runs A–C, of the effective-map and cross-readout runs and of the time-aligned clamps insert at t_{80}. Two analyses differ: the insertion-time control inserts at $t_{80} - 4$ (clipped at 0), t_{80}, $t_{80} + 4$, $t_{80} + 8$, 27 and 40, of which we compare t_{80} with update 40 (Appendix G.9), and the reuse experiment continues cached states computed for t_{80} or $3t_{80}$ updates (Appendix G.7).

We keep the pairs whose sham and full transplant both read the target at lag 0: 43–58 per checkpoint at $L = 8$ (830 in total) and 33–63 at $L = 6$ (829 in total). These pairs enter Runs A–C and the effective-map, cross-readout and time-aligned-clamp runs. The other analyses use the subsets or samples stated with them: the insertion-time control, the direction battery, the propagator and the fate maps select their own both-correct pairs, and the reuse experiment has no such filter. A pair *errs* if any of lags 1–20 reads a wrong label. Of the 16 predictions of Tables 26–27 that can be re-scored on lags 1–8 instead, only PK1 changes verdict (Table 27); the later predictions of Table 28 were not re-scored this way.

Run A labels and the insertion quantities of Section 4.1 come from logits saved in half precision; all later runs record labels at run time. At insertion the seam-only transplant changes the target margin by a median of 0.03–0.16 logits per roster checkpoint, while the half-precision spacing at the size of the target logit is 0.008–0.031; in up to 19% of pairs the change is exactly zero at half precision. The smallest seam-only values of $|c_0|$ are therefore resolved only to within a few spacings; a single-precision recomputation needs a GPU rerun and was not done.

Unless stated otherwise, intervals are 95% percentile intervals from bootstraps over contexts with checkpoints held fixed. They do not cover variation over training seeds, of which each task has three to five. Intervals from Runs A–C resample contexts within each checkpoint ($B = 10{,}000$; $4{,}000$ for medians); pooled statistics are stratified by checkpoint. The seeds of one task share their evaluation

contexts, so pooled intervals may be too narrow; where a re-analysis resamples each context jointly across the seeds of its task, we say so. A separate re-analysis of the effective-map and clamp runs (by other agents of the same model family, so not independent in the sense of a different implementer) first reported 90% intervals; every interval we quote from it was recomputed at 95% with the same context clusters (2,000 draws), so the main text and this appendix give each such statistic at one level. The insertion-time control keeps its 90% intervals, stated with each of them (Appendix G.9).

G.2 Interventions

Write $h[P]$ for the rows of a state at the positions in P, and $h_{S,t}$ for the sham trajectory.

Run A: position-restricted transplants ($L = 8$ only). The start is h_S with $h_S[P]$ replaced by $h_P[P]$, for P equal to all positions (*full*), $\{1, 2\}$ (*seam only*), $\{3, \ldots, L\}$ (*downstream only*) or $\{L+1\}$ (*answer only*). Two controls carry none of the donor's content. *Random* adds $\|v\|_2 \; u / \|u\|_2$ to h_S, with u standard Gaussian over the whole state, the start position included; we draw four u and report the largest error rate. *Other* h_0 is the recipient's own problem, computed for t_{80} updates from a different native initial state.

Run B: trajectory clamps. A clamp overwrites positions with the sham trajectory, $h_t[j] \leftarrow h_{S,t}[j]$, at every update $t = 0, \ldots, 35$. On the seam-only transplant we clamp one intermediate position $j \in \{3, \ldots, L\}$ at a time, all intermediate positions together, or the answer position; on the full transplant we clamp only the answer position.

Run C: necessity, dose, clamp-release and readout patches. *All but seam* is the full transplant with $h[\{1, 2\}]$ reset to $h_S[\{1, 2\}]$. *Dose* sets $h_S + a\, v$ on positions $\{1, 2\}$ only, for $a \in \{0.25, 0.5, 0.75, 1, 1.25, 1.5, 2\}$; $a = 1$ is the seam-only transplant. *Clamp-release* holds the answer position of the full transplant at the sham trajectory for states $0, \ldots, k$ and frees it afterwards ($k = 4, 8, 12$). *Late cut* clamps position L from state k on ($k = 0, 4, 8$). *Readout patches* leave the dynamics untouched. At lag t we evaluate the readout on the transplant's state with position j taken from the sham (*remove* j), and on the sham's state with position j taken from the transplant (*keep* j). At the full transplant's first-error lag, removing j *fixes* the error if the target returns, and keeping j *reproduces* it if the same wrong label appears.

Effective map of the answer position (EM). Attention is causal, so positions $0, \ldots, L$ never read the answer position: the recurrence is a skew product in which the upstream positions $u_t = h_t[\{0..L\}]$ drive the answer position $a_t = h_t[L+1]$ and receive nothing back. For a trajectory c and update t we freeze $u_{c,t}$, iterate only the answer position $K = 40$ times from $a_{c,t}$ (resetting the upstream after every update), and call the limit $a^*_c(t)$; $\ell^*_c(t)$ is the readout of $[u_{c,t}, a^*_c(t)]$. The iteration counts as converged when the answer position's relative change at step K is below 10^{-3}. We probe the sham, the seam-only transplant and the full transplant at lags 0–24. At every such lag of the two transplants we also start the frozen iteration from five points $a_{S,t} + \mu(a_{c,t} - a_{S,t})$, $\mu \in \{0, 0.25, 0.5, 0.75, 1\}$, and from the transplant's answer position under the sham's upstream (the *swap*); EM3 and the swap statistic read these at the seam-only transplant's first-error lag, and the one-label count of Appendix G.5 uses every lag. In native runs (the recipient's own computation from its native initial state, 32 contexts per checkpoint) we apply the same probe at every update and record the recipient's prefix products for EM5. For each pair we also record the two one-operand hybrid products $H_1 = d_1 a_2 \cdots a_L$ and $H_2 = a_1 d_2 a_3 \cdots a_L$, with (d_1, d_2) the donor's edited operands. The probe is counterfactual: frozen-upstream states do not occur on the model's trajectories.

Cross readouts (EM-X1). With $R(u, a)$ the readout of the state with upstream u and answer position a, we evaluate at every lag 0–24 $R(u_S, a^*_c)$ and $R(u_c, a^*_S)$ on the fixed points, and $R(u_S, a_c)$ (*ansact*) and $R(u_c, a_S)$ (*upact*) on the actual states; X1a and X1b read them at the seam-only transplant's first-error lag, and X1c at the lags up to it. Ansact wrong means the transplant's own answer-position state carries the wrong label; upact wrong means the exit blocks read the wrong label from the upstream positions directly.

Time-aligned clamps (G1c). For each pair a first pass rolls out the sham and the seam-only transplant and records, for every intermediate position j, the lag $\tau_j \in \{0, \ldots, 20\}$ at which that position's distance from the sham peaks. Label-free and per pair, this timing is taken from the pair's own unclamped run. A *window* clamps position $j \in \{3, M, L\}$, $M = L/2 + 1$, to the sham for the three states $\tau_j + d - 1, \ldots, \tau_j + d + 1$, with $d \in \{-4, 0, +4\}$, clipped at state 0. An early window lying entirely before lag 0 is empty and excluded; one clipped to state 0 alone is kept, although it changes nothing, because M is not displaced at insertion (79/470 roster pairs at L=8, 245/455 at L=6). *Hold* clamps the position at every update, as in Run B. *Transmission* replaces position M at every update by $h_S + b\,(h - h_S)$, $b \in \{0, 0.25, 0.5, 0.75\}$ ($b = 1$ is the seam-only transplant, $b = 0$ is holding M). Removal is $1 -$ rate(arm)/rate(seam only).

What holds by construction. Because attention is causal, no position changes because of positions to its right. Positions left of the edit therefore never move: their displacement from the sham is exactly 0 at every lag in every Run A transplant of donor states (full, seam only, downstream only, answer only). In Runs B and C, whose sham reference was rolled out in a separate batch, it is at most 2.2×10^{-4}, below the 0.008 difference between two batchings of the sham itself. For the same reason, a downstream-only transplant never moves positions 0–2, and clamping the answer position leaves positions $0, \ldots, L$ exactly as in the unclamped run. The seam also carries most of the displacement at insertion: its median share of the squared lag-0 displacement is 0.57–0.97 per checkpoint, over the roster and over all 16 checkpoints alike. Part of "the seam suffices and the downstream positions do not" thus follows from the construction; the all-but-seam condition tests necessity separately. As a check, the sham with its answer position clamped reads the same label as the sham at every lag in all 32 checkpoint–length cells.

Five further results follow from the construction (EM3 once the frozen map has a unique fixed point), whatever else the model computes, and we report them as consistency checks. (i) E1: after a prompt edit, positions left of the edit equal those of the unedited run. (ii) EM3: with the upstream frozen, every start whose iteration converges ends at a fixed point of the frozen map, so if that fixed point is unique, all five starts share its label and "the upstream decides" is expected. Wherever all five starts converged they reached one label (Appendix G.5); the probe saved labels, not end states, so uniqueness itself is not tested, and contraction is not guaranteed: a power-iteration estimate of the largest singular value of the frozen map's Jacobian at its fixed point, which bounds it from below, is below one in only 0.56 [0.50, 0.62] of roster cells (X2a, Table 28), so at most that share contracts in one step. EM3 is therefore a consistency check of the probe. (iii) EM4 and X1c: the fixed point at update t is computed from u_t, whereas the state a_t was computed from u_{t-1}, so the fixed point turns wrong one update before the state even if the state follows it as fast as possible. (iv) The shifted G1c windows: an early window mostly clamps a position that the disturbance has not yet reached, and a late window mostly starts at or after the first error (Appendix G.6). (v) GA0: replaying the undelayed difference at a position rewrites that position with the seam-only run's own state, so it must reproduce the seam-only labels up to batching precision. Separately, an answer-position clamp (P2) fixes a position the readout uses, so P2 is partly built in.

G.3 Pre-declared predictions: verdicts and provenance

Tables 26 and 27 score the 24 predictions that were written into analysis scripts before those scripts' first outputs existed. Each rule is paraphrased from the docstring with its thresholds and counts unchanged, and each clause is counted separately on the pre-declared chain roster of 9 checkpoints (A_5 s0, s1, s3, s4, D_{31} s0, s1, s4 and $\mathbb{Z}_{61}$ s0, s4), fixed before Runs B and C from an edit-position sweep. That sweep was later found to use a different pair filter (Appendix G.13). A re-analysis of the sweep with the both-correct filter compared seven candidate rules for the path type: thresholds on the slope of the peak error lag (two settings) and of the first-error lag, two rules on its bootstrap interval, a variant with a mixed class, and the stability of the type over bootstrap draws (at least 0.8); each rule is applied at both lengths and in both error windows (lags 1–8 and 1–20). Only 4 of the nine are chain-type under all of them with room to spare (A_5 s1, s3, s4 and D_{31} s4, the *robust four*). A_5 s0 passes the stability rule only at its threshold (0.802 against 0.8) and was left out; D_{31} s0 misses it (0.796); the others fail an interval or slope rule. The robust four were fixed at 13:02 from the sweep alone, after the outputs of Runs A and B had been seen and before Run C; they are a post hoc sensitivity set, and the scored counts use the declared roster. Over these two tables, 10 predictions held at both lengths, 5 held in part, 5 were not met and 4 hold by construction. A

prediction counts as held in part if it held at one length only, or if its first clause held and a later one failed; a clause that holds nearly by construction does not earn that status (EM2). Three counts need a qualification: P2 is counted as held although it is partly built in (it clamps an input of the readout); P1 is held in part only through its $L{=}6$ count, outside its declared scope, and at its declared length it is inconclusive (5 of 9, short of 7 and not below the falsifying 5); and X1a holds exactly at its threshold. The predictions are not independent (Pd's two misses are P2's; X1a–X1c were written after the effective-map outputs), so the counts are not a success rate. Table 28 scores the two later scripts the same way, outside these counts (below).

Counting choices. PK3 has two clauses; counted clause by clause, as its docstring states, both hold at both lengths (9 and 7 of 9 at $L{=}8$, 9 and 6 at $L{=}6$). The first analysis script instead required both clauses per checkpoint, which gives 7 and 6 of 9; both counts are reported. PK1 counted per checkpoint gives the same numbers as clause by clause (5 and 6). PK3's transmission b is nominal: the displacement that actually passes position M is smaller (Appendix G.6). In a context bootstrap of the Run C criteria (2,000 draws per checkpoint and length), the counts were met in 0.99/1.00 of draws for Pa's main clause, 0.79/0.66 for Pb, 1.00/1.00 for Pb′ and 0.97/1.00 for Pd ($L{=}8$/$L{=}6$), and Pc's pooled share reached 0.60 in 0.09/0.30; the other predictions were not re-analysed this way.

Dose and amplitude. Pb's monotone clause holds without its allowed dip in 9 of 9 checkpoints at $L{=}8$ but in only 7 at $L{=}6$: there D_{31} s1 and $\mathbb{Z}_{61}$ s4 dip once, by less than the allowed 0.05. Without the allowance the whole rule holds in 7 and 6 of 9, the latter below its threshold of 7. Pb′ compares median first-error lags at two nearby amplitudes and is coarse. Within pairs, regressing each pair's first-error lag on $\log_2 a$ ($a = 0.5$–2, pairs erring at three or more doses) shows that larger displacements err earlier. At $L{=}8$ the interval excludes zero in 7 of 9 roster checkpoints, all in that direction, where doubling a advances the first error by 0.10–1.13 updates. The other two intervals include zero: D_{31} s1 -0.02 [-0.64, 0.63] and $\mathbb{Z}_{61}$ s4 0.11 [-0.87, 0.70]. In A_5 s3 the paired fourfold contrast ($a = 2$ minus $a = 0.5$) is flat (-0.06 [-0.21, 0.09]). At $L{=}6$, 6 intervals exclude zero, 5 of them in the same direction (advances 0.10–1.67), while $\mathbb{Z}_{61}$ s4 errs later at larger amplitude (1.12 [0.77, 1.46]). The time of the error therefore depends on the amplitude; a disturbance carried to a threshold would also arrive earlier when larger, so these slopes do not separate carrying from growth.

Provenance. Table 30 lists, for every scored script, when the predictions were written, the SHA-256 prefix of the kept copy of the script file (docstring included; which version each copy holds is stated in the caption), when its outputs appeared and what was already known. P1–P3 are in the docstring of the Run B script, written at 10:52 JST (session log). At 10:52, before any output, a device cast was added to the code; at 11:15, after the $L = 8$ outputs (10:54–11:07) and before the $L = 6$ outputs (11:17–11:28), a fallback for the $L = 6$ insertion update was added. Neither edit touched the docstring, whose first version was recovered from the session log. P1 was stated for $L = 8$ only, so its $L = 6$ count applies the same rule outside its declared scope. The reuse experiment's amendment A1 (corrected rationale for E5, and a constant-wait comparator for E4) was recorded at 14:28 (session log; the script's own note corrects an earlier estimate of about 15:10), after all $L = 8$ outputs (the last at 14:24) and the first $L = 6$ output (14:25) had been written on the GPU host and, by the same log, before any output was downloaded or opened; the E1–E5 texts are unchanged. For EM1–EM5 and PK1–PK3, the two scripts' files were written at 17:50 and 17:51. Before any run, the PK1 docstring was amended once (17:51:30, session log) to exclude empty early windows; both scripts were then copied (17:51:35), before a smoke run on four pairs of one checkpoint (17:51:47) and before the first full outputs (17:52 and 17:53). X1a–X1c were written at 18:16, after the effective-map outputs (the last at 18:14) and their first analysis had been seen; they were written before their own run, but not blind to the effective-map results. The robust four were fixed at 13:02, after the outputs of Runs A and B and before Run C, the time-aligned clamps and the effective-map runs. The copies of the two later scripts were made before their runs. The timed-clamp script ran as written. The delay-line script as run differs from its copy only in the power iteration: 10 steps instead of 15, and 8 pairs at $L{=}8$ (none at $L{=}6$) instead of 16, a change that a note in the script dates after a smoke run and before the full run; its prediction texts are unchanged.

What was known when the predictions were written. None of these predictions is independent of earlier data. All were written after Experiment II's outcomes and after the edit-position sweep

	Rule as written (checkpoint criterion; count required)	L=8	L=6	robust four	By construction	Outcome
P1	Clamping any single intermediate position (every update) leaves $\leq 0.25\times$ the seam-only error; $\geq$ 7 of 9 at L=8 (falsified if < 5)	5/9	7/9	4, 4	–	held in part: L=8 mixed (short of 7, not below 5); L=6 met, outside the declared scope
P2	Clamping the answer position leaves $\leq 0.25\times$ the full transplant's error; $\geq$ 7 of 9	7/9	7/9	3, 3	partly: clamps a readout input	held; misses A_5 s3, D_{31} s1; the clamp left $> 0.5\times$ in 1 of 9 at either length (A_5 s3), not the ≥ 5 that P2 named as a sign of upstream reading
P3	In the five checkpoints declared direct or diffuse, clamping all intermediate positions keeps $\geq 0.5\times$ the seam-only error; $\geq$ 3 of 5	2/5	2/5	–	–	not met
Pa	All-but-seam error $\leq 0.25\times$ full; $\geq$ 5 of 9. Named exceptions A_5 s0, D_{31} s0, $\mathbb{Z}_{61}$ s0 stay $\geq 0.5\times$	5/9; 2/3	5/9; 2/3	4, 4	–	held in part: main clause at its minimum; exception clause failed (D_{31} s0: 0.49, 0.26); unnamed miss $\mathbb{Z}_{61}$ s4
Pb	Seam-only error non-decreasing in the dose $a \leq 1.5$ (one dip ≤ 0.05 allowed) and error at a=0.5 $\leq 0.5\times$ that at a=1; $\geq$ 7 of 9	7/9	7/9	3, 3	–	held; misses A_5 s3, D_{31} s1; at L=6, $\mathbb{Z}_{61}$ s4 passes only by the allowed dip
Pb′	Among pairs erring at a=1 and a=1.5, median first-error lags differ by ≤ 1; $\geq$ 7 of 9	9/9	9/9	4, 4	–	held by this coarse rule; within pairs, larger doses bring the first error slightly earlier
Pc	At the full transplant's first error, keeping only the answer position reproduces the wrong label in $\geq 60\%$ (pooled, roster without A_5 s3); in A_5 s3, keeping position L does so more often	0.558; held	0.583; held	–	–	not met: pooled share below 0.60 at both lengths; A_5 s3 clause held
Pd	Releasing an answer-position clamp after state 4 leaves $\geq 0.5\times$ the full error, after state 12 $\leq 0.25\times$; $\geq$ 6 of 9	7/9	7/9	3, 3	–	held; misses A_5 s3, D_{31} s1
E1	After a prompt edit at element j, positions $0..j$ equal those of the run on the unedited prompt at every lag (relative difference $\leq 10^{-4}$), under warm start and prefix reset	9/9	9/9	4, 4	yes: causal attention	consistency check (largest difference 1.3×10^{-6} in the roster, 1.5×10^{-6} over all 16)
E2	Answer-changing edits: slope of the warm start's settling lag against j in $[-1.5, -0.5]$ in $\geq$ 6 of 9; settles before scratch for $j \geq L/2$ in $\geq$ 6 of 9	9/9; 9/9	9/9; 9/9	4, 4	–	held
E3	Answer-preserving warm starts: slope of the first-error lag against j in $[-1.5, -0.5]$; $\geq$ 6 of 9	9/9	9/9	4, 4	–	held; the same 16 contexts as the edit-position sweep, with a new draw of k only
E4	Label-free stop $R(8,\ L{+}1{-}j)$ has $\leq 2\%$ wrong stops in $\geq$ 7 of 9, while $R(8, 0)$ has $> 5\%$ for preserving edits at $j \leq 1$ in ≥ 5 of 9	9/9; 0/9	9/9; 0/9	0, 0	–	held in part: first clause met, also against a constant wait of similar mean, which is equally safe; second clause failed (at most 0.031; mis-specified)
E5	Prefix reset is never less accurate than scratch at a matched lag, and settles no later than scratch in $\geq$ 8 of 9	1/9; 9/9	1/9; 9/9	0, 1	–	not met: accuracy falls below scratch at some lag (by up to 0.10); settling clause met

Table 26: **Pre-declared predictions (1 of 2): clamps, doses, readout patches and state reuse** [X]. Cells count the pre-declared roster checkpoints (A_5 s0, s1, s3, s4, D_{31} s0, s1, s4, $\mathbb{Z}_{61}$ s0, s4) that meet each clause of the rule, clauses separated by semicolons; pairs are those whose sham and full transplant both read the target at lag 0, and a pair errs if any of lags 1–20 reads a wrong label. The rules are paraphrased from the script docstrings, with thresholds and counts unchanged. "Robust four" (A_5 s1, s3, s4, D_{31} s4) counts checkpoints meeting every checkpoint-level clause (Pa: its main clause) at L=8 and L=6, as a sensitivity analysis. "By construction" marks clauses that hold whatever the model computes; we report those as consistency checks, not evidence. P3 counts the five checkpoints it names; Pc gives the pooled share, then the A_5 s3 clause. E1–E5 pool both cache times and all rows of the reuse experiment, without the both-correct filter; the verdicts are the same at either cache time. P1 was stated for L=8 only; at L=8 it is inconclusive, and it is held in part only through its L=6 count. SHA-256 prefixes of the first-written script files (docstring included), recovered from the session log at 14:25 (timing in Table 30): `d272dec7` (P1–P3), `982a1414` (Pa–Pd), `bcbe95dd` (E1–E5).

	Rule as written (checkpoint criterion; count required)	L=8	L=6	robust four	By construction	Outcome
EM1	Sham control: the frozen-upstream iteration converges and ℓ^* reads the target in $\geq 95\%$ of cells; ≥ 14 of 16	16/16	16/16	–	–	held (control)
EM2	ℓ^* turns wrong at or before the first error in $\geq 80\%$ of erring pairs, and stays right at lags 0–20 in $\geq 70\%$ of non-erring pairs; each ≥ 7 of 9	9/9; 2/9	9/9; 2/9	0, 0	first clause, nearly	not met: second clause failed (ℓ^* also turns wrong in 0.13–1.00 of non-erring pairs at L=8); the first clause needs only the built-in one-update lead: asked two updates ahead, it holds in 0 of 9 at either length
EM3	At the first error, all five answer-position starts end on one label in $\geq 80\%$ of erring events; ≥ 7 of 9	9/9	9/9	4, 4	yes, given a unique fixed point: skew product	consistency check; one end label, end states not saved; not a one-step contraction (X2a: 0.56 of cells; spectral radius not measured); under the sham's positions $0..L$ the transplant's answer-position state returns to the target in 0.93–1.00 of events
EM4	Median lead (first-error lag minus first lag with ℓ^* wrong) ≥ 0; ≥ 7 of 9	9/9	9/9	4, 4	yes: one-update offset	consistency check (median lead 0.0–2.0)
EM5	Native runs, before ℓ^* first reads the answer: (a) ℓ^* is a prefix product at $\geq 3\times$ chance with median normalised entropy ≤ 0.5; (b) median normalised entropy ≥ 0.5; each ≥ 7 of 9	(a) 0/9; (b) 2/9	(a) 0/9; (b) 2/9	(b) 0, 0	–	neither reading; (b) only in A_5 s0, D_{31} s0
PK1	At the middle position M, a 3-update clamp aligned with the disturbance's peak removes $\geq 60\%$ of the seam-only error in ≥ 6 of 9, while the windows shifted by ± 4 remove $\leq 30\%$ in ≥ 6 of 9	5/9; 9/9; 9/9	6/9; 9/9; 9/9	4, 4	shifted-window clause, partly	held in part: met at L=6 only; misses D_{31} s0, D_{31} s1, $\mathbb{Z}_{61}$ s0, $\mathbb{Z}_{61}$ s4 (L=8); shifted windows remove -0.03 to 0.08 and 0.00 to 0.14 (all 16); with lags 1–8, 6 of 9 at L=8
PK2	The aligned window removes $\geq 0.8\times$ what clamping M at every update removes; ≥ 6 of 9	6/9	5/9	4, 4	–	held in part: met at L=8 only
PK3	Error non-decreasing in the transmission b at M (one dip ≤ 0.05) in ≥ 7 of 9, and half the seam-only rate reached at some $b \in (0, 1)$, not at b=0, in ≥ 6 of 9	9/9; 7/9	9/9; 6/9	3, 2	–	held; half clause at its threshold at L=6; b is nominal, the effective transmission is lower
X1a	At the first error, the transplant's own answer-position state read with the sham's positions $0..L$ is wrong in $\geq 60\%$ of events; ≥ 4 of the 6 checkpoints whose answer position Pc found necessary	4/6	4/6	1, 1 of 2	–	held at its threshold; misses A_5 s4, $\mathbb{Z}_{61}$ s4
X1b	A_5 s3: the transplant's positions $0..L$ read with the sham's answer-position state are wrong in $\geq 60\%$ of events, the reverse in $\leq 20\%$	0.85; 0.04	0.86; 0.04	–	–	held
X1c	The answer position's frozen-upstream fixed point reads wrong ≥ 1 update (median) before its state does; ≥ 4 of 6	6/6	6/6	2, 2 of 2	yes: one-update offset	consistency check; beyond the built-in update, 1 and 0 of 6

Table 27: **Pre-declared predictions (2 of 2): the answer position's effective map, time-aligned clamps and cross readouts** [X]. Cells, pairs, "robust four" and "by construction" as in Table 26. $\ell^*(t)$ is the label read after freezing positions $0..L$ at update t and iterating only the answer position; $M = L/2 + 1$. X1a and X1c count the six checkpoints they name. X1a–X1c were written before their run but after the effective-map outputs had been seen. The aligned windows are timed from the same pair's unclamped run. SHA-256 prefixes of the script files, copied before their runs: `6c5ae3ab` (EM1–EM5), `7dc2a56d` (PK1–PK3), `a7e18997` (X1a–X1c). Over both tables, of 24 predictions 10 held at both lengths (P2 partly by construction; X1a at its threshold), 5 held in part (at one length only, or in the first clause but not a later one; P1 only through its L=6 count, outside its declared scope), 5 were not met (EM5 counted here) and 4 hold by construction. The predictions are not independent, so these counts are not a success rate. Of the 16 predictions that can be re-scored with lags 1–8 instead of 1–20, only PK1 changes (it is then met at L=8 as well). Two later scripts with predictions stated before running (GA0–GA2, GC1–GC3; R3a–R3b, X2a–X2b) were analysed after this tally was compiled (Table 28).

	Rule as written (checkpoint criterion; count required)	L=8	L=6	robust four	By construction	Outcome
GA0	Replay check: delaying by k=0 reproduces the seam-only transplant's labels in $\geq 99\%$ of (pair, lag) cells, in every checkpoint	16/16 (min 1.00)	16/16	–	yes: rewrites the run's own state	consistency check
GA1	At M, overwriting the difference with its value k updates earlier shifts the median first error (pairs erring with and without the delay) into $[k-1, k+1]$, for k=2 and k=4; ≥ 5 of 9	8/9	9/9	4, 4	–	held; exactly k in 6 and 8 of 9; the L=8 miss $\mathbb{Z}_{61}$ s4 shifts when the delay is at position L instead
GA2	In the 7 other checkpoints, the median shift for k=4 at M is ≤ 1; ≥ 4 of 7	6/7	6/7	–	–	held; exception $(\mathbb{Z}_2)^6$ s4 (shift 4.0)
GC1	Sham states, all 16 checkpoints: median σ_1 of the 32-update map <1 in ≥ 14 of 16, and $\log \sigma_1/32 < 0$ in $\geq 90\%$ of states pooled	15/16; 0.91	–	4	–	held as run; pooled clause just above its threshold (58 of 64 states), below it on the both-correct pairs the preamble names (0.89); lower bounds favour this verdict; exception $(\mathbb{Z}_2)^6$ s4 (median 14.6)
GC2	Descriptive: share of states with σ_1 of the 8-update map >1, per checkpoint	$\geq$half: 15/16; all: 12/16	–	all: 4	–	description (finite-window amplification); all states in all 9 roster checkpoints
GC3	Median σ_1 of the 8-update map at the transplant state $\leq 1.5\times$ that at the sham state; ≥ 12 of 16	12/16	–	4	–	held at its threshold; misses D_{31} s0, s1, $\mathbb{Z}_{61}$ s4, $(\mathbb{Z}_2)^6$ s1
R3a	Robust four, declared for L=8: removal by a 3-update clamp of M centred at $\tau_M + d$ is largest at $d \in \{-1, 0\}$, and at d=+2 at most half that largest; ≥ 3 of 4	4/4	4/4	(the set)	–	held; counted strictly (ties with d=+1 not credited) 3 and 1 of 4; L=6 outside the declared length
R3b	Robust four: a window centred at the median τ_M of the checkpoint's other pairs removes $\geq 0.8\times$ the d=0 window's removal; ≥ 3 of 4	4/4	4/4	(the set)	–	held; still needs other pairs' unclamped runs and the sham trajectory
X2a	Roster, pooled: largest singular value κ of the frozen answer map's Jacobian at its fixed point <1 in $\geq 95\%$ of (pair, lag) cells	0.56	–	–	–	not met; medians above one in A_5 s0 (2.2), D_{31} s0 (1.6) and D_{31} s1
X2b	Answer-route four, pooled: where the fixed point reads wrong, median ratio of the answer state's distance to it after and before one update in $(0.2, 0.8)$	0.46	–	–	–	held (n=155): the state closes about half the gap per update

Table 28: **Pre-declared predictions of two later scripts: delay line, maximal growth, timed clamps and the frozen answer map** [X]. Written into the delay-line script (GA, GC) and the timed-clamp script (R3, X2) before their first runs, after the outputs behind Table 27 had been seen, and analysed after the tally of Tables 26–27 had been compiled; they are kept out of that tally. Cells as in Table 26, except where a rule names another set: GA0 all 64 pairs and lags 0–35; GA2 the seven other checkpoints; GC all 16 checkpoints at L=8 only, on the first 8 pairs without the both-correct filter; R3 the robust four; X2a pooled roster cells and X2b pooled cells of A_5 s0, D_{31} s0, s4 and $\mathbb{Z}_{61}$ s0 (the X1a hits), both at L=8 and on the first 16 pairs without the both-correct filter. GA1 counts the rule's band; at k=2 the band includes a shift of one, which GA2 counts as none. GC and κ come from power iteration, which bounds the largest singular value from below: an estimate above one shows growth, an estimate below one does not prove contraction. GC counts each of the 4 distinct sham states per checkpoint once, by the larger of its two estimates; counting the 8 computed pairs gives the same verdicts. τ_M is the pair's own label-free peak lag of the seam-only difference at M. Of the 9 predictions (GC2 is descriptive), 7 held (GC1's pooled clause just above its threshold and not met on both-correct pairs; GC3 at its threshold), 1 was not met and 1 holds by construction; they are not independent, so these counts are not a success rate. SHA-256 prefixes of the copies made before the first runs: `a1085d30` (GA, GC; its power iteration was reduced before the full run, Appendix G.13) and `2474fd64` (R3, X2).

that fixed the roster. The Run B docstring cites Run A's seam-only result. Pa–Pd were written after Runs A and B: Pa's named exceptions are, in the docstring's own words, the checkpoints whose downstream-only transplant errs in Run A; Pd's two misses (A_5 s3, D_{31} s1) are P2's two misses; and Pc's clause for A_5 s3 was written after Run B had shown that its answer clamp leaves most of the error. E3 was written after the edit-position sweep and re-estimates its slope on the same 16 contexts with a new draw of the edit element k (Appendix G.7). X1a names the six checkpoints whose readout patches (Pc) had marked the answer position necessary. The delay-line script, requested by the writing plan, was written after all of these outputs; the timed-clamp script was written after the re-analysis that found PK1's shifted windows uninformative, and it names the robust four as its set. The thresholds, the window width of three updates, the offsets ± 4 and ± 2 and $K = 40$ were set by the analysing agent (Claude) before the runs; the authors did not set or review them before the runs. We report the rules so that every miss is visible, not as confirmations.

Separately, automated "theory" analyses recorded numeric predictions, mostly for Run B, in 14 hashed files. Every one was completed after the first $L = 8$ output of Run B existed; we do not count them as predictions and do not report their scores. Four GPU scripts reported in Appendices G.9 and G.10 (the direction battery, the insertion-time control, the propagator and the fate maps) also stated predictions before running, but the statistics or operational options that decide them were not fixed before running (for the fate maps the authors chose one option afterwards, Appendix G.10), so we report them as exploratory.

Two later scripts (Table 28). Two scripts stated predictions in their docstrings before their first run (copies at 19:06 and 19:19, SHA-256 `a1085d30` and `2474fd64`; first outputs 19:10 and 19:23). Both were written after the effective-map and clamp outputs had been seen, and the second after a re-analysis of those runs that proposed its tests. They were analysed (20:01 and 20:06) after the tally above had been compiled (19:25), so Table 28 scores them by their rules as written but keeps them out of that tally: of their 9 predictions, 7 held (GC1 only on the pairs as run; below), 1 was not met and 1 holds by construction. Pairs are both-correct and errors are wrong labels in lags 1–20 unless stated.

Delay line (roster). The seam-only transplant and the sham are first rolled out freely. The delay arm starts from the seam-only transplant and overwrites position $M = L/2 + 1$ at every update by the sham's state plus the seam-only transplant's difference at that position k updates earlier (zero before update k); all other positions run freely. For both $k = 2$ and $k = 4$, the median shift of the first error, over pairs that err in both arms (11–48 per checkpoint and k at L=8), lies in the rule's band $k \pm 1$ in 8 of 9 checkpoints at L=8 and 9 at L=6 (GA1, predicted in at least 5); it equals k exactly in 6 and 8. The L=8 miss, $\mathbb{Z}_{61}$ s4, shifts by 0.0 and 1.0 ($k = 2, 4$) at M but by 2.0 and 4.0 when the delay is applied at position L (descriptive). The delay mostly moves errors rather than removing them: over $k = 2$ and 4 it changes the error rate by at most 0.11 (L=8) and 0.14 (L=6; $\mathbb{Z}_{61}$ s4 at $k = 2$, where the rate halves), and a pair whose delayed error leaves lags 1–20 drops out of the median. Per pair (post hoc, pooled over the roster), 80–85% of pairs erring in both arms shift by $k \pm 1$ and 12–15% not at all, so the medians hide a minority of errors that bypass M. At $k = 2$ the band includes a shift of one, which GA2 counts as none. A shift of about k is what any error that must pass M shows, whether the difference is carried or grows downstream; the delay line shows that most errors pass M, not that they are carried rather than amplified. In the seven other checkpoints the shift for $k = 4$ is at most one in 6 at either length (GA2, predicted in at least 4); the exception, $(\mathbb{Z}_2)^6$ s4, shifts by 4.0. The undelayed replay (GA0) reproduces the seam-only labels by construction.

Maximal growth (L=8, all 16 checkpoints). A power iteration on $J^\top J$, with J the Jacobian of the t-update map ($t = 1, 4, 8, 16, 32$) at the sham state at insertion, was run for the first 8 pairs per checkpoint with 10 steps (the docstring specified 16 pairs and 15 steps; Appendix G.13). The two edits of a context share the sham state, so these are 4 distinct states per checkpoint (64 in all), each estimated twice from different random starts. The two estimates differ by up to 0.24 of the larger one at $t \leq 8$ but by at most 0.04 at $t = 16$ and 32, so the iteration had not always converged at short horizons; every value bounds σ_1 from below. Counting each state once, by its larger estimate: the median σ_1 of the 32-update map is below one in 15 of 16 checkpoints, and $\log \sigma_1/32 < 0$ at 58 of the 64 states (0.91, against a required 0.90; GC1 held, with the same verdict when each computed pair is counted). The sample has no both-correct filter although the docstring's preamble names both-correct pairs; restricted to the 106 both-correct pairs among them, the pooled share is 0.89 (94 of 106; the same on distinct states), below 0.90, while the first clause still holds in 15 of

16. GC1's verdict thus depends on the pair filter, and estimates from below favour it. An estimate below one does not prove contraction. On the roster the 32-update estimate is below one at 34 of 36 distinct states; the other two, both in A_5 s0, exceed one, which as lower bounds shows growth over 32 updates there. In $(\mathbb{Z}_2)^6$ s4, off the roster, the median stays near 14.6 from 8 to 32 updates, a direction that does not contract and was not examined further. The 8-update map's σ_1 exceeds one at every state in 12 of 16 checkpoints, including all 9 roster checkpoints, and at half of them or more in 15 (GC2, descriptive; per-checkpoint medians over the computed pairs 0.41–14.6, roster 1.8–12.9). At transplant states the median σ_1 of the 8-update map is at most 1.5 times the sham's in 12 of 16 (GC3, exactly at its threshold; ratios 0.76–5.47).

Timed clamps at M (robust four; Table 29). A first pass records, for each of the checkpoint's 64 pairs, the lag $\tau_M \in \{0, \ldots, 20\}$ at which the seam-only transplant's distance from the sham at M peaks (label-free). A window clamps M to the sham trajectory for three updates centred at $\tau_M + d$, $d \in \{-2, \ldots, 2\}$. R3a, declared for L=8: the removal is largest at $d \in \{-1, 0\}$, and at $d = +2$ it is at most half that largest value, in 4 of 4 checkpoints (predicted in at least 3); at L=6, outside the declared length, in 4. The analysis took the first maximum. In 1 checkpoint at L=8 and 3 at L=6 the largest removal is shared with $d = +1$; counting only checkpoints where it is not, R3a holds in 3 of 4 at L=8 (still met) and 1 at L=6. Unlike PK1's windows shifted by four updates, the $d = +2$ window starts before the first wrong readout in 0.93–1.00 of erring pairs per checkpoint at L=8 (0.96–1.00 at L=6; post hoc), so its small removal is not forced by the timing: windows that contain the peak update remove most of the error, and at L=8 windows that miss it by one update ($d = \pm 2$) remove at most half as much. R3b: a window centred at the median τ_M of the checkpoint's other pairs (leave-one-out) removes at least 0.8 times what the window at $d = 0$ removes in 4 of 4 at L=8 and 4 at L=6 (predicted in at least 3). It uses no timing from the pair's own run, but it needs the other pairs' unclamped runs, and every window writes the sham trajectory into M. The robust four include none of the four roster checkpoints in which holding M leaves more than a quarter of the error (D_{31} s0, s1, $\mathbb{Z}_{61}$ s0, s4 at L=8; Appendix G.6), so R3 tests timing only where M is nearly a cut point.

Contraction of the frozen answer map (roster, L=8). For the first 16 pairs of each checkpoint (without the both-correct filter named in the docstring's preamble; Appendix G.13) and lags 0–20 of the seam-only transplant, the answer position's frozen-upstream map is iterated 40 times toward a fixed point, and 8 power-iteration steps estimate the largest singular value κ of its Jacobian at the end point, again from below. $\kappa < 1$ in only 0.56 [0.50, 0.62] of the 3,024 cells (median 0.90 [0.807, 0.996]; X2a predicted at least 0.95, not met). Per-checkpoint medians range from 0.70 to 2.2 and are below one in 6 of 9; they exceed one in D_{31} s1 and in the answer-route checkpoints A_5 s0 (2.2) and D_{31} s0 (1.6). This script did not record convergence; in the effective-map run the same frozen iteration on the both-correct pairs' transplants converged in all but 7 of 23,500 probed cells at L=8 (Appendix G.5). If it converges here too, a one-step gain above one points to a non-normal Jacobian; its spectral radius was not measured. Where the end point reads wrong, the answer state's distance to it after one update is a median 0.46 [0.43, 0.47] of its distance before, so the state closes 54% [53, 57] of the gap per update ($n = 155$ cells of the four answer-route checkpoints, per-checkpoint medians 0.39–0.50; X2b predicted a ratio in $(0.2, 0.8)$, held). Restricted to both-correct pairs (9–16 of the 16 per checkpoint), the two statistics are 0.57 and 0.46 ($n = 129$), with the same verdicts. The X2 intervals are 95% context-cluster bootstraps computed after the run (the script reported none).

	window centred at $\tau_M + d$					leave-one-out	hold M
Checkpoint	d=−2	−1	0	+1	+2	(R3b)	(every update)
A_5 s1	0.21/0.14	0.86/1.00	0.93/1.00	0.86/1.00	0.43/0.36	0.93/1.00	0.93/1.00
A_5 s3	0.02/0.00	0.73/0.76	1.00/1.00	1.00/1.00	0.04/0.04	1.00/1.00	1.00/1.00
A_5 s4	0.21/0.09	0.76/0.82	0.76/0.94	0.58/0.91	0.08/0.09	0.74/0.94	0.79/0.91
D_{31} s4	0.07/0.07	0.78/1.00	1.00/1.00	0.93/1.00	0.15/0.33	1.00/1.00	1.00/1.00

Table 29: **Timed clamps at the middle position** [X], robust four, cells L=8/L=6: removal 1 − rate(arm)/rate(seam only), errors in lags 1–20, both-correct pairs. τ_M is the pair's own label-free peak lag of the seam-only difference at M; a window clamps M to the sham trajectory for three updates. The leave-one-out window is centred at the median τ_M of the checkpoint's other pairs. No intervals were computed; the seam-only transplant errs in 14–48 pairs per checkpoint at L=8 and 14–51 at L=6.

Predictions (script)	written (JST)	SHA-256	first output	Known when written
P1–P3 (Run B)	10:52 (log)	`d272dec7`	10:54	Experiment II; edit-position sweep; Run A outputs
Pa–Pd, Pb′ (Run C)	13:24	`982a1414`	13:37	also Run B outputs
E1–E5 (reuse)	13:37 (log)	`bcbe95dd`	13:41	Runs A and B outputs, Run C starting; amendment A1 at 14:28, before any output was opened
EM1–EM5 (EM)	17:50 (file)	`6c5ae3ab`	17:52	also the reuse experiment, the insertion-time and direction controls, and the re-analyses of Runs A–C, the propagator and the fate maps; robust four fixed at 13:02
PK1–PK3 (G1c)	17:51 (file)	`7dc2a56d`	17:53	as for EM
X1a–X1c (EM-X1)	18:16 (file)	`a7e18997`	18:17	also the EM and G1c outputs (last 18:14) and their first analysis
GA0–GC3 (delay line)	19:06 (file)	`a1085d30`	19:10	also the X1 outputs, the 33-direction growth bound (Appendix G.10) and files of the re-analysis of the EM and G1c runs; power iteration changed before the full run (text)
R3a–X2b (timed clamps)	19:19 (file)	`2474fd64`	19:23	also that re-analysis, which proposed these tests

Table 30: **Provenance of the scored predictions** [X]. "Log": session log of the automated pipeline; "file": file time of the script. SHA-256 prefixes are of the copies kept with the outputs (docstring included). The copies for P1–P3, Pa–Pd and E1–E5 are the first-written versions, recovered from the session log at 14:25, after the outputs of Runs B and C and of the reuse experiment at L=8 and before its first L=6 output; later edits to these scripts are stated in the text. The copies for EM, PK, X1 and the two later scripts (GA–X2, scored separately in Table 28) were made before their first runs and include any code edits made before copying. All times are from 2026-09-25, recorded by the same pipeline that ran the analyses, and none is externally time-stamped; that no output was seen before a prediction was written is self-reported.

G.4 Complete per-checkpoint results of Runs A–C

Tables 31–35 give every checkpoint at both lengths (Run A: L=8 only), and Table 36 gives the edit-position sweep. The nine roster checkpoints come first; the robust four are in bold. A dagger marks a checkpoint where a pre-declared checkpoint-level criterion was missed. Cells read "L=8/L=6". Ratios are error rates relative to the unclamped condition named in the column head and can exceed one; "–" in the path-type column means not typed.

Several checkpoints qualify the main-text summary. In A_5 s0 the seam-only transplant errs more often than the full transplant (0.94 against 0.49, Cohen's $\kappa_C = 0.04$ at L=8), and its downstream-only transplant errs on its own (0.67), as do those of D_{31} s0 and $\mathbb{Z}_{61}$ s0. At L=6 the full transplant of $\mathbb{Z}_{61}$ s0 errs more often than its seam-only transplant (difference +0.26 [+0.16, +0.38]). In A_5 s3 the answer clamp leaves most of the error, and the readout patches point to position L instead. In D_{31} s1 part of the error survives every clamp. In $\mathbb{Z}_{61}$ s1 an answer-only transplant errs in 0.49 of pairs, and in $(\mathbb{Z}_2)^6$ s1 a random displacement errs in 0.60; in these two checkpoints an error does not need the donor's operand positions. Removing the answer position's difference from the readout input fixes 0.897–1.00 of first errors in six roster checkpoints at L=8 (at least 0.90 in 5 of the nine); at L=6 it fixes at least 0.90 in 7 of the nine.

The statistics of Section 4 in full. Section 4 states its results with fewer numbers and without prediction labels; this paragraph gives them in full (roster, both-correct pairs, errors in lags 1–20, L=8 unless stated). *Where the difference starts.* In 8 of the 9 roster checkpoints the seam-only transplant errs about as often as the full transplant: full minus seam-only rates are -0.02 to $+0.08$ (95% intervals within $[-0.11, +0.24]$; no equivalence margin was set; A_5 s0: -0.45). The answer position alone errs about as often as the sham (excess at most $+0.04$). All but the seam leaves at most 0.08 of the error in 5 of 9 at both lengths (Pa, held in part: D_{31} s0, one of three checkpoints named as exceptions expected to keep at least half, kept 0.49; $\mathbb{Z}_{61}$ s4, not named, kept 0.27). *Delaying and blocking it.* Under the delay line at M, the first errors of pairs that err in both runs shift by a median within $k\pm1$ for $k = 2$ and 4 in 8 of 9 roster checkpoints at L=8 (miss: $\mathbb{Z}_{61}$ s4) and 9 at L=6 (GA1, needing 5; exactly k in 6 and 8), while error rates change by at most 0.11 (0.14 at L=6). In 6 of the 7 other checkpoints the shift is at most one update at both lengths (GA2; $(\mathbb{Z}_2)^6$ s4 shifts by k). Any route through M would give a shift of about k, so its size speaks to neither speed nor amplification. Held alone at every update, each intermediate position removes at least 3/4 of the seam-only error in 5 of 9 (P1 at L=8: seven predicted, below five falsifying); P3 (direct or diffuse checkpoints) was

Checkpoint	n	sham	full	seam only $\{1,2\}$	κ_C full, seam	peak lag full/seam	downstream $\{3..L\}$	answer $\{L+1\}$	random (max of 4)	other h_0
Pre-declared chain roster										
A_5 s0	49	0.00	0.49	0.94	0.04	9/8	0.67	0.00	0.08	0.00
A_5 **s1**	51	0.02	0.25	0.27	0.95	8/8	0.02	0.02	0.02	0.02
A_5 **s3**	52	0.00	0.92	0.92	1.00	8/8	0.02	0.02	0.02	0.00
A_5 **s4**	57	0.00	0.68	0.67	0.96	7/7	0.02	0.00	0.04	0.00
D_{31} s0	49	0.00	0.76	0.67	0.70	7/7	0.37	0.04	0.12	0.00
D_{31} s1	52	0.00	0.83	0.79	0.88	9/9	0.04	0.00	0.02	0.00
D_{31} **s4**	55	0.00	0.47	0.49	0.89	8/8	0.02	0.02	0.00	0.00
$\mathbb{Z}_{61}$ s0	49	0.04	0.55	0.49	0.39	7/7	0.47	0.04	0.08	0.04
$\mathbb{Z}_{61}$ s4	56	0.00	0.27	0.27	0.73	9/9	0.07	0.00	0.02	0.00
Other checkpoints										
A_5 s2	43	0.00	0.40	0.42	0.95	8/6	0.07	0.00	0.02	0.00
$\mathbb{Z}_{61}$ s1	57	0.00	0.61	0.33	0.22	1/1	0.14	0.49	0.23	0.00
$\mathbb{Z}_{61}$ s3	51	0.00	0.65	0.57	0.75	2/1	0.16	0.16	0.04	0.00
$(\mathbb{Z}_2)^6$ s0	48	0.04	0.69	0.67	0.48	1/3	0.35	0.31	0.08	0.04
$(\mathbb{Z}_2)^6$ s1	58	0.00	0.78	0.84	0.78	5/5	0.26	0.34	0.60	0.00
$(\mathbb{Z}_2)^6$ s3	54	0.00	0.57	0.59	0.81	3/4	0.28	0.22	0.02	0.00
$(\mathbb{Z}_2)^6$ s4	49	0.04	0.51	0.39	0.43	2/8	0.24	0.08	0.12	0.04

Table 31: **Run A: position-restricted transplants** [X], L=8, share of pairs with any error in lags 1–20. κ_C is Cohen's kappa between the full and seam-only error indicators (not the singular value κ of X2a); "peak lag" is the lag with the largest per-lag error share (not the lag by which most pairs have erred). "Random" is a whole-state displacement of the donor's size; "other h_0" is the recipient's own problem from another initial state.

		error rate		clamp on seam-only / seam-only			answer clamp on	path type
Checkpoint	n	full	seam only	worst single mid (P1)	all mids (P3)	answer	full / full (P2)	(post hoc)
Pre-declared chain roster								
A_5 s0	49/49	0.49/0.51	0.94/0.96	0.07/0.04	0.00/0.00	0.00/0.00	0.00/0.00	S
A_5 **s1**	51/41	0.25/0.32	0.27/0.34	0.21/0.00	0.07/0.00	0.21/0.14	0.23/0.08	S
A_5 **s3**	52/56	0.92/0.91	0.92/0.91	0.06/0.02	0.00/0.00	0.88/0.88	$0.92^\dagger$/$0.90^\dagger$	U
A_5 **s4**	57/63	0.68/0.48	0.67/0.52	0.21/0.09	0.00/0.00	0.13/0.18	0.15/0.17	S
D_{31} s0	49/48	0.76/0.79	0.67/0.85	$0.39^\dagger$/$0.37^\dagger$	0.00/0.00	0.00/0.00	0.00/0.00	M
D_{31} s1	52/54	0.83/0.80	0.79/0.87	$0.49^\dagger$/$0.49^\dagger$	0.34/0.32	0.46/0.40	$0.44^\dagger$/$0.47^\dagger$	P
D_{31} **s4**	55/33	0.47/0.42	0.49/0.45	0.04/0.07	0.00/0.00	0.00/0.00	0.00/0.00	S
$\mathbb{Z}_{61}$ s0	49/61	0.55/0.46	0.49/0.20	$0.42^\dagger$/0.25	0.12/0.00	0.08/0.00	0.07/0.00	M
$\mathbb{Z}_{61}$ s4	56/50	0.27/0.26	0.27/0.28	$0.40^\dagger$/0.21	0.00/0.00	0.00/0.00	0.00/0.00	M
Other checkpoints (the P3 set is A_5 s2, $\mathbb{Z}_{61}$ s1, s3, $(\mathbb{Z}_2)^6$ s0, s3)								
A_5 s2	43/50	0.40/0.30	0.42/0.34	0.83/1.00	$0.11^\dagger$/$0.06^\dagger$	0.94/1.12	0.76/1.00	U
$\mathbb{Z}_{61}$ s1	57/61	0.61/0.77	0.33/0.48	0.84/0.79	$0.47^\dagger$/$0.21^\dagger$	0.00/0.03	0.03/0.00	B
$\mathbb{Z}_{61}$ s3	51/52	0.65/0.44	0.57/0.50	0.97/0.96	0.69/0.81	0.62/0.50	0.55/0.48	P
$(\mathbb{Z}_2)^6$ s0	48/56	0.69/0.73	0.67/0.77	0.84/0.93	$0.31^\dagger$/$0.30^\dagger$	1.00/0.88	0.79/0.85	–
$(\mathbb{Z}_2)^6$ s1	58/41	0.78/0.78	0.84/0.76	1.00/1.03	0.90/1.03	0.33/0.35	0.38/0.34	B
$(\mathbb{Z}_2)^6$ s3	54/58	0.57/0.71	0.59/0.71	0.94/0.93	0.56/0.80	0.12/0.17	0.03/0.22	B
$(\mathbb{Z}_2)^6$ s4	49/56	0.51/0.54	0.39/0.34	0.58/0.26	0.21/0.00	0.26/0.05	0.52/0.27	–

Table 32: **Run B: trajectory clamps** [X], cells L=8/L=6, lags 1–20. "Worst single mid" is the largest error left by clamping one intermediate position. Daggers: P1 (rule ≤ 0.25; declared for L=8, applied to L=6 descriptively), P2 (≤ 0.25), P3 (≥ 0.5). Path types summarise the L=8 columns of this table and were assigned after seeing them, so they are descriptions, not findings. S: serial chain to the answer position; M: several parallel lanes to the answer position; U: chain read away from the answer position (A_5 s3 through position L, Table 35; A_5 s2 through intermediate positions); B: seam-to-answer bypass; P: a direct component that persists under every clamp; –: not typed. At L=6, $\mathbb{Z}_{61}$ s0 and s4 meet the single-clamp rule (0.25, 0.21; the first exactly at the threshold), so their L=8 label does not carry over.

not met. *Timed clamps.* Clamping M for three updates around its difference peak in the pair's own seam-only run removes at least 60% of the error in 5 and 6 of 9 (lengths 8 and 6; PK1 needs six, so it is not met at L=8); the misses are checkpoints where holding M throughout removes only 0.56–0.71 (post hoc, L=8). In the robust four, removal is tuned in time: windows two updates before the peak (partly limited by construction) or after it remove 0.02–0.21 and 0.04–0.43, against 0.76–1.00

Checkpoint	all but seam / full (Pa)	seam-only error rate at dose a 0.25	0.5	0.75	1	1.25	1.5	2	Pb met
Pre-declared chain roster									
A_5 s0*	1.38/1.08	0.00/0.00	0.33/0.37	0.82/0.82	0.94/0.96	0.96/0.98	0.96/0.98	0.98/0.98	yes/yes
A_5 **s1**	0.08/0.00	0.02/0.00	0.06/0.00	0.14/0.15	0.27/0.34	0.67/0.59	0.76/0.68	0.78/0.73	yes/yes
A_5 **s3**	0.00/0.00	0.17/0.16	0.67/0.54	0.87/0.79	0.92/0.91	0.98/0.95	0.98/0.96	1.00/0.96	no/no
A_5 **s4**	0.03/0.00	0.00/0.00	0.19/0.16	0.51/0.33	0.67/0.52	0.74/0.63	0.77/0.70	0.84/0.76	yes/yes
D_{31} s0*	0.49†/0.26†	0.06/0.08	0.24/0.33	0.49/0.67	0.67/0.85	0.80/0.90	0.84/0.90	0.90/0.92	yes/yes
D_{31} s1	0.02/0.02	0.06/0.06	0.44/0.56	0.65/0.80	0.79/0.87	0.85/0.85	0.85/0.91	0.94/0.96	no/no
D_{31} **s4**	0.04/0.00	0.00/0.00	0.20/0.06	0.40/0.21	0.49/0.45	0.58/0.61	0.73/0.67	0.78/0.79	yes/yes
$\mathbb{Z}_{61}$ s0*	0.85/0.86	0.04/0.00	0.08/0.02	0.27/0.11	0.49/0.20	0.65/0.34	0.76/0.48	0.86/0.69	yes/yes
$\mathbb{Z}_{61}$ s4	0.27†/0.31†	0.00/0.00	0.05/0.04	0.20/0.22	0.27/0.28	0.36/0.26	0.52/0.32	0.57/0.46	yes/yes
Other checkpoints (no prediction)									
A_5 s2	0.24/0.27	0.00/0.00	0.12/0.08	0.26/0.24	0.42/0.34	0.51/0.56	0.65/0.64	0.74/0.72	yes/yes
$\mathbb{Z}_{61}$ s1	0.94/0.91	0.00/0.02	0.07/0.15	0.25/0.31	0.33/0.48	0.44/0.54	0.60/0.74	0.77/0.85	yes/yes
$\mathbb{Z}_{61}$ s3	0.76/0.87	0.04/0.06	0.22/0.19	0.41/0.38	0.57/0.50	0.61/0.54	0.69/0.62	0.71/0.73	yes/yes
$(\mathbb{Z}_2)^6$ s0	0.76/0.66	0.12/0.07	0.23/0.34	0.48/0.64	0.67/0.77	0.73/0.84	0.75/0.84	0.85/0.86	yes/yes
$(\mathbb{Z}_2)^6$ s1	0.60/0.66	0.28/0.22	0.64/0.63	0.76/0.73	0.84/0.76	0.83/0.76	0.86/0.78	0.86/0.80	no/no
$(\mathbb{Z}_2)^6$ s3	0.77/0.66	0.09/0.03	0.43/0.53	0.63/0.71	0.59/0.71	0.61/0.84	0.72/0.83	0.81/0.86	no/no
$(\mathbb{Z}_2)^6$ s4	0.52/0.60	0.04/0.00	0.06/0.05	0.20/0.18	0.39/0.34	0.63/0.48	0.69/0.66	0.82/0.77	yes/yes

Table 33: **Run C: seam necessity and dose** [X], cells L=8/L=6, lags 1–20. An asterisk marks the three checkpoints named in advance as exceptions to Pa (predicted ≥ 0.5); a dagger marks a Pa miss, of the main clause or of the exception clause. "Pb met" applies the Pb rule of Table 26, including its allowed dip of at most 0.05.

Checkpoint	Pb′: |Δ median first-error lag| (pairs)	answer clamped for states 0..k, / full k=4	k=8	k=12	position L clamped from state k, / full k=0	k=4	k=8
Pre-declared chain roster							
A_5 s0	0.0 (46)/0.0 (47)	0.96/0.96	0.71/0.00	0.00/0.00	0.00/0.08	0.08/0.12	0.58/1.00
A_5 **s1**	0.0 (14)/0.0 (14)	1.00/1.00	0.31/0.08	0.23/0.08	0.08/0.00	0.08/0.00	0.62/1.00
A_5 **s3**	0.0 (48)/0.0 (51)	1.00/1.00	0.92/0.90	0.92†/0.90†	0.00/0.02	0.00/0.02	0.88/1.00
A_5 **s4**	0.0 (36)/0.0 (33)	1.00/1.03	0.23/0.17	0.15/0.17	0.08/0.13	0.15/0.20	0.85/1.00
D_{31} s0	0.0 (32)/0.0 (40)	1.00/1.00	0.59/0.24	0.05/0.00	0.30/0.26	0.30/0.66	0.95/1.00
D_{31} s1	0.5 (38)/0.0 (44)	1.00/1.02	0.81/0.51	0.44†/0.47†	0.44/0.53	0.47/0.53	0.63/1.02
D_{31} **s4**	0.0 (27)/0.0 (15)	1.00/1.00	0.38/0.00	0.00/0.00	0.04/0.00	0.04/0.00	0.92/1.00
$\mathbb{Z}_{61}$ s0	0.5 (24)/1.0 (12)	0.89/0.89	0.44/0.00	0.07/0.00	0.22/0.11	0.33/0.71	0.93/1.00
$\mathbb{Z}_{61}$ s4	0.5 (14)/1.0 (11)	1.00/1.00	0.67/0.00	0.00/0.00	0.40/0.08	0.47/0.08	0.67/0.92
Other checkpoints (no prediction)							
A_5 s2	0.0 (17)/1.0 (17)	0.94/1.07	0.76/1.07	0.76/1.00	0.82/1.13	0.82/1.00	0.88/1.00
$\mathbb{Z}_{61}$ s1	0.0 (19)/1.0 (29)	0.40/0.34	0.03/0.02	0.03/0.00	0.86/0.91	0.86/0.96	1.00/1.00
$\mathbb{Z}_{61}$ s3	0.0 (29)/0.0 (26)	0.70/0.52	0.70/0.65	0.70/0.52	0.91/0.91	1.03/1.00	1.00/1.00
$(\mathbb{Z}_2)^6$ s0	0.0 (32)/1.0 (43)	0.79/0.85	0.79/0.85	0.79/0.85	0.88/0.85	0.88/0.88	1.00/1.00
$(\mathbb{Z}_2)^6$ s1	1.0 (48)/1.0 (31)	0.93/0.78	0.49/0.53	0.42/0.47	1.00/1.00	0.98/1.00	0.98/1.00
$(\mathbb{Z}_2)^6$ s3	1.0 (31)/0.0 (39)	0.55/0.54	0.19/0.32	0.03/0.22	0.71/0.85	0.84/0.93	0.97/0.98
$(\mathbb{Z}_2)^6$ s4	5.0 (15)/1.0 (19)	0.92/0.73	0.56/0.27	0.52/0.27	0.44/0.50	0.68/0.70	1.00/1.00

Table 34: **Run C: amplitude, clamp-release and late cut** [X], cells L=8/L=6, lags 1–20. The first column compares seam doses a=1 and a=1.5 among pairs that err at both (Pb′ requires ≤ 1). Pd requires ≥ 0.5 at k=4 and ≤ 0.25 at k=12 (daggers mark the failing entry). The late cut was not part of any prediction.

for the best window (L=8: R3a in 4 of 4, needing 3; 3 if ties with the next window are not credited; L=6, not declared: 4, strictly 1). Timed from the checkpoint's other pairs, the window removes 0.74–1.00, at least 0.96 times as much (R3b in 4 and 4 of 4 at lengths 8 and 6). Partial clamps at M raise the error gradually (PK3, held). *Where it is read.* The transplant's answer-position state, read with the sham's upstream positions, gives most wrong labels in 4 of the 6 checkpoints predicted to route the difference through the answer position (X1a, met at its threshold); the reverse holds in A_5 s3 (0.85 against 0.04; X1b). Tables 37 and 38 give the per-checkpoint values, and Appendix G.5 the fixed-point analyses.

Checkpoint	first errors	answer position $L{+}1$ keep: reproduces	answer position $L{+}1$ remove: fixes	last operand position L keep: reproduces	last operand position L remove: fixes
Pre-declared chain roster					
A_5 s0	24/25	0.96/0.92	1.00/1.00	0.00/0.00	0.04/0.00
A_5 **s1**	13/13	0.08/0.08	0.85/1.00	0.08/0.00	0.92/0.92
A_5 **s3**	48/51	0.00/0.04	0.12/0.12	0.52/0.76	0.94/0.94
A_5 **s4**	39/30	0.21/0.30	0.90/0.90	0.03/0.03	0.54/0.57
D_{31} s0	37/38	1.00/0.97	1.00/1.00	0.00/0.00	0.00/0.00
D_{31} s1	43/43	0.30/0.35	0.58/0.58	0.00/0.00	0.26/0.14
D_{31} **s4**	26/14	0.65/0.86	1.00/1.00	0.00/0.00	0.27/0.21
$\mathbb{Z}_{61}$ s0	27/28	0.78/0.68	0.93/1.00	0.07/0.00	0.22/0.29
$\mathbb{Z}_{61}$ s4	15/13	0.33/0.23	1.00/1.00	0.00/0.00	0.60/0.69
Other checkpoints					
A_5 s2	17/15	0.06/0.00	0.29/0.13	0.06/0.07	0.18/0.13
$\mathbb{Z}_{61}$ s1	35/47	0.86/0.79	0.97/1.00	0.00/0.00	0.06/0.04
$\mathbb{Z}_{61}$ s3	33/23	0.18/0.35	0.55/0.61	0.00/0.00	0.06/0.04
$(\mathbb{Z}_2)^6$ s0	33/41	0.45/0.12	0.42/0.22	0.12/0.20	0.27/0.39
$(\mathbb{Z}_2)^6$ s1	45/32	0.78/0.69	0.82/0.69	0.02/0.03	0.07/0.03
$(\mathbb{Z}_2)^6$ s3	31/41	0.74/0.61	0.97/0.85	0.03/0.02	0.10/0.22
$(\mathbb{Z}_2)^6$ s4	25/30	0.80/0.87	0.56/0.77	0.32/0.00	0.24/0.07

Table 35: **Run C: readout patches at the full transplant's first error** [X], cells $L{=}8/L{=}6$. "Keep" evaluates the readout on the sham state with one position taken from the transplant; "remove" evaluates it on the transplant state with that position taken from the sham. The dynamics are not changed. Pc predicted the pooled keep-answer share and, for A_5 s3, that keeping position L reproduces errors more often than keeping the answer position; the rest is descriptive. A patch keeping L and the answer position together was not run; the cross readouts of Table 38 separate the two routes differently.

Checkpoint	both-correct filter $L{=}8$	both-correct filter $L{=}6$	sham-only filter $L{=}8$
Pre-declared chain roster			
A_5 s0	-1.18 [-1.21, -0.71]	-1.00 [-1.20, -1.00]	-1.18
A_5 **s1**	-1.11 [-1.18, -0.93]	-1.10 [-1.30, -0.81]	-1.11
A_5 **s3**	-0.96 [-1.11, -0.86]	-1.20 [-1.20, -1.00]	-0.96
A_5 **s4**	-0.89 [-1.00, -0.75]	-1.00 [-1.10, -0.90]	-0.89
D_{31} s0	-1.14 [-1.21, -0.82]	-0.90 [-1.30, -0.60]	-1.14
D_{31} s1	-0.93 [-1.14, -0.82]	-1.20 [-1.40, -0.20]	-0.89
D_{31} **s4**	-1.21 [-1.21, -1.00]	-1.00 [-1.30, -1.00]	-1.21
$\mathbb{Z}_{61}$ s0	-0.86 [-1.25, 0.18]	-0.70 [-0.90, -0.30]	-0.96
$\mathbb{Z}_{61}$ s4	-0.57 [-0.66, -0.29]	-0.40 [-0.60, -0.10]	-0.57
Other checkpoints			
A_5 s2	-0.04 [-0.75, 0.04]	0.00 [-0.70, 0.20]	-0.39
$\mathbb{Z}_{61}$ s1	0.25 [-0.14, 0.32]	0.00 [-0.30, 0.40]	0.25
$\mathbb{Z}_{61}$ s3	-0.00 [-0.14, 0.07]	0.00 [-0.20, 0.60]	-0.00
$(\mathbb{Z}_2)^6$ s0	-0.07 [-0.57, 0.39]	0.20 [-0.80, 0.20]	0.04
$(\mathbb{Z}_2)^6$ s1	-0.32 [-0.61, 0.11]	-0.20 [-1.10, -0.10]	-0.43
$(\mathbb{Z}_2)^6$ s3	0.07 [-0.68, 0.39]	-0.40 [-0.80, 0.20]	0.07
$(\mathbb{Z}_2)^6$ s4	-0.39 [-1.00, 0.11]	0.10 [-1.10, 0.30]	-0.36

Table 36: **Edit-position sweep** [X]: OLS slope over the edited element j of the earliest peak error lag (lags 1–20), in updates per position, with 95% context-bootstrap intervals. 16 contexts per checkpoint (blocks 0–1, a subset of the 32 contexts of Runs A–C) with one random k each; answer-preserving edit $(a_j k, k^{-1} a_{j+1})$ for $j = 0, \ldots, L-2$, transplanted at t_{80}. Both-correct: sham and edit-j donor read the target at lag 0; the last column uses the sham-only filter of the sweep that fixed the roster (Appendix G.13). The robust four (bold) were selected by type rules applied to this sweep's slopes and intervals (Appendix G.3; the slopes alone do not single them out), so their slopes are not independent evidence for the chain type. The reuse experiment's E3 (Appendix G.7) re-estimates the slope on the same 16 contexts with a new draw of k, not on new contexts, with the roles of the cached and edited prompts reversed and without the both-correct filter.

G.5 THE ANSWER POSITION'S EFFECTIVE MAP AND CROSS READOUTS

Tables 37 and 38 give the per-checkpoint values behind EM2–EM5 and X1a–X1b (EM1 is the sham control; X1c is summarised in Table 27). Of these, only EM2's second clause, EM5 and X1a–X1b are informative beyond the construction (Appendix G.2). EM2's second clause failed: in non-erring pairs ℓ^* also turns wrong, in 0.54 [0.48, 0.61] of them pooled over the roster at $L{=}8$ (0.48 [0.42, 0.54] at $L{=}6$), so a wrong fixed point is not a warning that the readout will err. EM5 found neither reading. Before the native effective map first reads the answer, its label is a prefix product at 0.08–0.15 per checkpoint, against a chance rate of 0.11 ($L{=}8$), and its median normalised entropy is 0.10–0.81; only A_5 s0 and D_{31} s0 exceed 0.5, at both lengths. In the other seven checkpoints the median entropy is below 0.5: at most 0.31 except D_{31} s1 (0.49, just below the threshold) at $L{=}8$, and at most 0.41 at $L{=}6$; in none is the label a running product. What a low-entropy label encodes is not identified by this probe.

Post hoc re-analysis. A separate re-analysis of the saved arrays (by other agents of the same model family; intervals recomputed at 95% over context clusters) adds four descriptions. They are post hoc and exploratory, and they probe a counterfactual state. (i) In every probed cell where all five starts converged, they reached one label: 0/23,493 cells with more than one label at $L{=}8$ and 0/22,726 at $L{=}6$ (roster; the 7 of 23,500 cells at $L{=}8$ and 24 of 22,750 at $L{=}6$ in which some start did not converge are excluded). The probe saved end labels, not end states, so this shows one end label per cell (five starts on one segment, roster only), not a single fixed point. (ii) In the four checkpoints whose readout reads the transplant's own answer position (A_5 s0, D_{31} s0, s4, $\mathbb{Z}_{61}$ s0; X1a), the readout errs at update t almost only if $\ell^*(t{-}1)$ was already wrong: among updates whose readout was still right at $t-1$, the error probability is 0.001 [0.000, 0.002] when $\ell^*(t{-}1)$ was right and 0.558 [0.489, 0.630] when it was wrong ($L{=}8$; 0.002 [0.000, 0.003] and 0.613 [0.557, 0.672] at $L{=}6$). (iii) In those checkpoints the readout's wrong label equals $\ell^*(t{-}1)$'s in 0.79 [0.74, 0.84] of cells, against 0.04 under a permutation null (0.76 [0.71, 0.81] and 0.07 at $L{=}6$). (iv) Both return: over the roster, 285/286 erring pairs read the target again at lag 24 at $L{=}8$ and 274/274 at $L{=}6$, and ℓ^* returns in the same pairs. In these four checkpoints, the results describe the transplant as moving the end label of the answer position's frozen-upstream map (the same from all five starts on the probed segment) to a wrong label and back. When the readout errs, it does so one update after ℓ^* has turned wrong, the earliest the update order allows, but it does so in only about half of such updates, and the state closes only part of its distance to the end point of the frozen-upstream iteration per update, a median 54% [53, 57] at $L{=}8$ (X2b, Appendix G.3). The fixed point gives no reliable warning beyond the built-in update: asked two updates ahead, EM2's first clause holds in 0 of 9 checkpoints at $L{=}8$ and 0 at $L{=}6$, although the median lead reaches two updates in D_{31} s4 (both lengths) and $\mathbb{Z}_{61}$ s4 ($L{=}8$; Table 37). The description does not extend to A_5 s3, which errs with no delay through the upstream positions (X1b), nor to the robust checkpoints A_5 s1 and s4, where neither cross readout alone reproduces more than half of the errors.

Route and transit (post hoc). Route does not determine whether the delay at M moves first errors. With the same cross readouts on all 16 checkpoints (a route counts when its part alone reproduces at least 0.6 of first errors), the answer position alone does so in 8 (A_5 s0, D_{31} s0, s4, $\mathbb{Z}_{61}$ s0, s1, $(\mathbb{Z}_2)^6$ s1, s3, s4), the upstream positions in 3 (A_5 s2, s3, $(\mathbb{Z}_2)^6$ s0) and neither in 5, identically at both lengths. Each route occurs both where the delay at M shifts first errors by about k (median per-pair shift within $k \pm 1$ for $k = 2$ and 4, at both lengths) and where it does not: A_5 s3 versus s2 for the upstream route, A_5 s0 versus $\mathbb{Z}_{61}$ s1 for the answer route.

G.6 TIME-ALIGNED CLAMPS

Table 39 gives the time-aligned clamps for all 16 checkpoints. At the middle position M, a three-update clamp timed to the peak of the pair's own disturbance removes 0.76–1.00 of the seam-only error in the robust four at $L{=}8$ (0.94–1.00 at $L{=}6$), as much as holding M at every update (0.79–1.00 and 0.91–1.00). In these four checkpoints both are near their ceiling, so the equality shows that position M is nearly a cut point there (some displacement still bypasses it; Effective transmission, below), not that the disturbance is confined to three updates. On the whole roster the timed clamp removes 0.47–1.00 ($L{=}8$), and holding M removes 0.56–1.00; in D_{31} s0, s1 and $\mathbb{Z}_{61}$ s0, s4 even holding M leaves more than a quarter of the error at $L{=}8$, so the disturbance also passes by other routes. Here the timing comes from the same pair's unclamped run. A later script removes that need

Checkpoint	n	erring pairs	EM2 (i): ℓ^* wrong by first error	EM2 (ii): ℓ^* right throughout (n)	EM3: one label	swap restores	EM4: median lead	EM5: native entropy
A_5 s0	49/49	46/47	1.00/1.00	0.67 (3)/0.50 (2)	1.00/1.00	1.00/1.00	1.0/1.0	0.59/0.53
A_5 s1	51/41	14/14	1.00/1.00	0.11 (37)/0.26 (27)	1.00/1.00	0.93/1.00	1.0/1.0	0.20/0.21
A_5 s3	52/56	48/51	1.00/1.00	0.00 (4)/0.20 (5)	1.00/1.00	1.00/1.00	0.5/0.0	0.30/0.20
A_5 s4	57/63	38/33	1.00/1.00	0.16 (19)/0.33 (30)	1.00/1.00	1.00/1.00	1.0/1.0	0.21/0.20
D_{31} s0	49/48	33/41	0.97/0.95	0.88 (16)/1.00 (7)	1.00/1.00	1.00/1.00	1.0/1.0	0.81/0.66
D_{31} s1	52/54	41/47	0.95/0.96	0.27 (11)/0.14 (7)	1.00/1.00	1.00/1.00	1.0/1.0	0.49/0.41
D_{31} s4	55/33	27/15	1.00/1.00	0.36 (28)/0.39 (18)	1.00/1.00	1.00/1.00	2.0/2.0	0.17/0.13
$\mathbb{Z}_{61}$ s0	49/61	24/12	0.96/1.00	0.56 (25)/0.63 (49)	1.00/1.00	1.00/1.00	1.0/1.5	0.20/0.18
$\mathbb{Z}_{61}$ s4	56/50	15/14	1.00/1.00	0.83 (41)/0.81 (36)	1.00/1.00	1.00/1.00	2.0/1.0	0.10/0.04

Table 37: **Effective map of the answer position** [X], pre-declared roster, cells L=8/L=6. n: both-correct pairs; erring: pairs whose seam-only transplant errs in lags 1–20. EM2 (i): share of erring pairs with ℓ^* wrong at some lag up to the first error. EM2 (ii): share of non-erring pairs with ℓ^* right at every lag 0–20, with their number in parentheses. EM3: share of erring events in which all five starts reach one label. Swap: share in which the transplant's answer position under the sham's upstream reaches the target. EM4: median of first-error lag minus the first lag with ℓ^* wrong. EM5: median normalised entropy of the native effective map's label before it first reads the answer. EM2 (i), EM3, the swap and EM4 hold nearly or entirely by construction. The robust four are A_5 s1, s3, s4 and D_{31} s4.

Checkpoint	erring events	ℓ^* wrong (self)	ansact wrong	upact wrong
A_5 s0	46/47	0.96/0.94	1.00/0.98	0.00/0.00
A_5 s1	14/14	0.93/0.86	0.14/0.07	0.07/0.07
A_5 s3	48/51	0.98/0.96	0.04/0.04	0.85/0.86
A_5 s4	38/33	0.95/0.91	0.50/0.48	0.08/0.09
D_{31} s0	33/41	0.88/0.76	1.00/1.00	0.00/0.00
D_{31} s1	41/47	0.85/0.81	0.32/0.47	0.41/0.38
D_{31} s4	27/15	0.67/0.40	0.74/0.80	0.00/0.00
$\mathbb{Z}_{61}$ s0	24/12	0.38/0.83	0.75/0.83	0.08/0.00
$\mathbb{Z}_{61}$ s4	15/14	0.80/0.86	0.53/0.07	0.00/0.00

Table 38: **Cross readouts at the seam-only transplant's first error** [X], pre-declared roster, cells L=8/L=6. Ansact: the transplant's own answer-position state read with the sham's positions $0..L$; upact: the transplant's positions $0..L$ read with the sham's answer-position state. X1a predicted ansact ≥ 0.6 in at least four of A_5 s0, s4, D_{31} s0, s4 and $\mathbb{Z}_{61}$ s0, s4 (met in four; $\mathbb{Z}_{61}$ s4 differs between the lengths, 0.53 and 0.07, both misses); X1b predicted upact ≥ 0.6 and ansact ≤ 0.2 in A_5 s3. Outside the roster, A_5 s2 (upact 0.89, ansact 0.00) and $(\mathbb{Z}_2)^6$ s0 (0.84, 0.25) also read the upstream (L=8). In A_5 s1 and s4 neither route alone reproduces more than half of the errors.

(R3b, Table 29): a window centred at the median peak time of the checkpoint's other pairs removes 0.74–1.00 of the error in the robust four at L=8, at least 0.96 times what the pair's own timing removes. It still needs the other pairs' unclamped runs, and every clamp writes an uncontaminated sham trajectory into M, so these are intervention analyses, not deployable defences.

The shifted windows are largely uninformative by construction. The windows shifted by ± 4 updates remove -0.03 to 0.08 at M in all 16 checkpoints at L=8 (0.00 to 0.14 at L=6), but this cannot show that the disturbance is localised in time. In the roster, 74% of pairs at L=8 (86% at L=6) have an early window that is empty or carries less than a tenth of the pair's peak displacement at M, so there is little to clamp. Among erring pairs, the late window starts at or after the first error in 70% (96%) and strictly after it in 28% (65%). A window starting after the first error cannot prevent it, and one starting at that lag can act on it only through position M's own contribution to the readout at that lag, so in most erring pairs the late window cannot prevent the error. PK1's shifted-window clause therefore holds almost regardless of the model (ledger, Appendix G.13). A later script shifts the window by one and two updates instead (R3a, Table 29). There the window two updates late still starts before the first wrong readout in 0.93–1.00 of erring pairs (L=8), yet it removes only 0.04–0.43 of the error against a largest removal of 0.76–1.00 per checkpoint: in the robust four, windows that contain the peak update at M remove most of the error, and windows that miss it by one update remove at most half as much. This bears on localisation in time at M; it does not rescue PK1's clause, which stays uninformative.

Effective transmission. PK3 scales the displacement at M by b at every update, but position M feeds back into itself, so the attenuation compounds. Measured as the displacement at M summed over lags 1–20 relative to the unclamped run (median over pairs), the effective transmission at nominal $b = 0.25$, 0.5 and 0.75 is 0.19–0.22, 0.40–0.45 and 0.65–0.70 per roster checkpoint at L=8 (0.18–0.22, 0.38–0.45, 0.64–0.70 at L=6). It is monotone in b, so PK3's verdict does not change when b is read as the effective transmission. Holding M throughout still leaves a displacement of 0.06–0.53 of the unclamped one at the answer position (L=8), a route that bypasses M; in multi-lane checkpoints the threshold in b is therefore confounded with it. For example, A_5 s3's error rate at $b = 0, 0.25, 0.5, 0.75, 1$ is 0.00, 0.10, 0.52, 0.81, 0.92.

Single-position clamps and error-time tracking (post hoc). Holding position 3 throughout removes 0.61–1.00 of the error in the roster at L=8 (0.63–1.00 at L=6). Among the seven other checkpoints it removes at most 0.28 in 4 at L=8 (a fifth, $(\mathbb{Z}_2)^6$ s0, removes 0.281) and in 5 at L=6; $(\mathbb{Z}_2)^6$ s1 removes nothing (0.00), but the largest removal among them is 0.67 (L=8) and 0.84 (L=6). To avoid the roster, which was chosen on error-time tracking, we correlated removal with the both-correct sweep slope over all 16 checkpoints. The rule was proposed by the analysing agent before this analysis ran (Spearman $\rho \leq -0.5$ at both lengths and a roster-minus-other difference whose 95% interval excludes zero), but after the per-checkpoint removals had been seen; it is not blind, and the authors adopted it after this analysis. The roster-minus-other difference inherits the roster's selection on error-time tracking, so it is partly circular. For holding position 3, $\rho = -0.75$ [-0.91, -0.39] at L=8 and -0.63 [-0.92, -0.14] at L=6 (checkpoint bootstrap); for the timed clamp at M, -0.87 [-0.96, -0.63] and -0.62 [-0.89, -0.16]. The roster-minus-other differences are 0.59 [0.40, 0.77] and 0.56 [0.28, 0.80] (hold) and 0.54 [0.36, 0.72] and 0.48 [0.21, 0.72] (timed). The rule is met, but L=6 has little margin: only 0.74 and 0.72 of bootstrap draws give $\rho \leq -0.5$, and resampling contexts within checkpoints as well raises the upper ends to -0.11 and -0.06. The relation is mostly a separation of two groups, not a graded one: within the roster, ρ is -0.60 to -0.27 at L=8 and -0.46 to -0.42 at L=6, and within the other checkpoints -0.32 to -0.18 (L=8) and 0.22 to 0.56 (L=6).

		seam-only	window at M (removal)			hold M	hold 3	PK1/PK2/PK3
Checkpoint	n	error rate	timed	4 early	4 late	(removal)	(removal)	met
Pre-declared chain roster								
A_5 s0	49/49	0.94/0.96	0.89/0.77	0.00/0.00	0.00/0.00	1.00/1.00	0.98/0.98	yyy/yny
A_5 s1	51/41	0.27/0.34	0.93/1.00	0.00/0.00	0.00/0.00	0.93/1.00	0.79/1.00	yyn/yyn
A_5 s3	52/56	0.92/0.91	1.00/1.00	0.00/0.00	0.00/0.00	1.00/1.00	1.00/1.00	yyy/yyy
A_5 s4	57/63	0.67/0.52	0.76/0.94	-0.03/0.00	0.00/0.00	0.79/0.91	0.92/0.97	yyy/yyy
D_{31} s0	49/48	0.67/0.85	0.48/0.32	0.00/0.00	0.03/0.00	0.67/0.66	0.61/0.63	nny/nny
D_{31} s1	52/54	0.79/0.87	0.59/0.55	0.05/0.00	0.00/0.00	0.56/0.60	0.66/0.66	nyy/nyy
D_{31} s4	55/33	0.49/0.45	1.00/1.00	0.00/0.00	0.04/0.00	1.00/1.00	1.00/0.93	yyy/yyn
$\mathbb{Z}_{61}$ s0	49/61	0.49/0.20	0.54/0.58	0.06/0.00	0.04/0.08	0.71/0.75	0.67/0.75	nnn/nny
$\mathbb{Z}_{61}$ s4	56/50	0.27/0.28	0.47/0.64	0.08/0.14	0.07/0.07	0.60/0.86	0.93/0.79	nny/ynn
Other checkpoints (no prediction)								
A_5 s2	43/50	0.42/0.34	0.33/0.12	0.06/0.00	0.00/0.00	0.44/0.18	0.67/0.76	nnn/nnn
$\mathbb{Z}_{61}$ s1	57/61	0.33/0.48	0.16/0.48	0.00/0.00	0.00/0.03	0.26/0.48	0.16/0.21	nnn/nyn
$\mathbb{Z}_{61}$ s3	51/52	0.57/0.50	0.07/0.08	0.00/0.00	0.00/0.00	0.07/0.08	0.07/0.12	nyn/nyn
$(\mathbb{Z}_2)^6$ s0	48/56	0.67/0.77	0.31/0.28	0.07/0.00	0.00/0.00	0.41/0.30	0.28/0.07	nnn/nyn
$(\mathbb{Z}_2)^6$ s1	58/41	0.84/0.76	0.02/0.00	0.00/0.00	0.00/0.00	0.02/-0.03	0.00/-0.03	nyn/nyn
$(\mathbb{Z}_2)^6$ s3	54/58	0.59/0.71	0.03/0.12	0.00/0.00	-0.03/0.00	0.06/0.12	0.13/0.12	nnn/nyn
$(\mathbb{Z}_2)^6$ s4	49/56	0.39/0.34	0.47/0.84	0.00/0.00	0.00/0.00	0.58/0.84	0.42/0.84	nyy/yyy

Table 39: **Time-aligned clamps** [X], cells L=8/L=6, lags 1–20; $M = L/2 + 1$ (position 5 at L=8, 4 at L=6). Removal is $1 -$ rate(arm)/rate(seam only) and can be negative. "Timed": three states around the peak of the pair's own seam-only displacement at M; "4 early/late": the same window shifted by ∓ 4 updates, largely uninformative by construction (Appendix G.6); empty early windows are excluded, and early windows of state 0 alone, which change nothing, are kept. The last column gives y/n for PK1, PK2 and PK3 at L=8/L=6, judged per checkpoint on all clauses (PK3's two clauses conjoined), so it can differ from the clause-wise counts of Table 27; for other checkpoints it is descriptive. The robust four are A_5 s1, s3, s4 and D_{31} s4.

G.7 REUSE AFTER A PROMPT EDIT

Design. The reuse experiment is exploratory [X]; its predictions E1–E5 were written before it ran. A looped model holds a cached state for a prompt A, computed from the native initial state for $t_{\text{cache}} \in \{t_{80}, 3t_{80}\}$ updates; the prompt is then edited at element j (operand position $j{+}1$) to give B. Under B we compare starts from the same native initial state: recomputing from scratch; continuing from A's cached state (a *warm start*); and the cached state with the positions right of the edit reset to their initial values (*prefix reset*, which joins states of update t_{cache} and update 0; Appendix G.13). An answer-preserving edit is $(a_j, a_{j+1}) \to (a_j k, k^{-1} a_{j+1})$ for $j = 0, \ldots, L-2$; an answer-changing edit is $a_j \to a_j k$ for $j = 0, \ldots, L-1$. A warm start after an answer-preserving edit resembles a transplant with the roles of the prompts reversed: the state computed for the unedited prompt continues under the edited one. We use 16 contexts (blocks 0–1) per checkpoint, the same contexts and native initial states as the edit-position sweep (18/18 checkpoint–length cells of the roster; the edit element k is newly drawn and differs in 283/288 checkpoint–length–context combinations), all 16 checkpoints at both lengths, and read labels at lags 0–60 without a both-correct filter. The *settling onset* s_8 is the first lag of the first run of eight consecutive readouts equal to B's answer; it equals $C_8 - 7$ in the notation of Section 2, where C_8 is the lag at which that run ends. The repeat rule $R(8, b)$ stops at the first lag $t \geq b + 7$ that ends eight identical readouts (at most lag 35), so $R(8, 0)$ is D_8; a stop is wrong if its label is not B's answer. Predictions are scored in Table 26; the analysis pools both cache times, and every verdict is the same at either.

Outcomes. E1 holds by construction (the largest relative difference is at float precision). E3 held: slopes of the first-error lag against j are -1.04 to -0.66 at $L{=}8$ and -1.15 to -0.50 at $L{=}6$; with $t_{\text{cache}} = t_{80}$ alone, 8 of 9 fall in the predicted range at $L{=}6$ (D_{31} s1 misses it). Because the contexts are those of the sweep, E3 is a re-estimate with new edits, not a replication on new data. E4(ii) was mis-specified: at $j \leq 1$ an answer-preserving warm start reads the right answer for about eight lags before its transient error, so $R(8, 0)$ stops correctly before the error arrives; the wrong stops occur after answer-changing edits (below). Under E4(i) the distance-aware wait $R(8, L{+}1{-}j)$ stops wrongly in at most 1.25% of warm starts per checkpoint at $L{=}8$ (in none at $L{=}6$). It stops wrongly exactly as often as the amendment's constant comparator, and on average only 0.1–0.8 updates earlier at $L{=}8$. The comparator waits $(L{+}4)/2$ updates. The amendment defined it as the mean of $L{+}1{-}j$, which is $(L{+}4)/2$ over answer-preserving edits but $(L{+}3)/2$ over answer-changing ones, so "a constant wait of equal mean" is approximate. Waiting $L{+}1$ updates, the distance-aware wait at the first edit position, gives the same per-checkpoint wrong-stop rates as the distance-aware wait (post hoc): at most 1.25% of warm starts over all edits and 1.56% over answer-changing edits at $L{=}8$, and none at $L{=}6$ (roster, both cache times).

Description (post hoc). Pooled over the roster, for answer-changing edits, the warm start's median settling onset falls from 10 at $j = 0$ to 3 at $j = L{-}1$ for $L{=}8$ (scratch: 11), and from 8 to 3 for $L{=}6$ (scratch: 9). In the C_8 notation these are 17 to 10 against 18, and 15 to 10 against 16. D_8 stops such a warm start on a wrong answer in 53.8%, 31.3% and 12.5% of edits at $j = 0, 1, 2$ for $L{=}8$, and in 7.6% and 1.7% at $j = 0, 1$ for $L{=}6$. Most of these stops are on the pre-edit answer (the unedited prompt's answer): 51.4%, 28.5% and 10.8% of edits at $L{=}8$ and 7.6% and 1.7% at $L{=}6$. Waiting $L{+}1{-}j$ updates before the repeat rule leaves at most 0.7% wrong stops per edit position. Experiment II's edits start at the first operand ($j = 0$); its larger answer-changing wrong-stop rates under D_8 at $L{=}8$ than at $L{=}6$ fit this pattern, but this is a description, not a test on Experiment II's pairs.

G.8 EDIT-AWARE REUSE: FULL TABLE, STATISTICS AND BOUNDARIES

Section 6 re-analyses the saved runs of the reuse experiment (Appendix G.7), which drew new edits on the contexts that selected the edit-ordered checkpoints, and Experiment II's; it makes no new runs. The first edited operand position p counts from 1, so the wait $L{+}2{-}p$ is the wait $L{+}1{-}j$ of Appendix G.7 with $p = j{+}1$. Unless stated, results are for answer-changing edits on the edit-ordered checkpoints, with both cache times pooled. The pre-edit answer is the unedited prompt's answer, recovered from the same context's answer-preserving edits. Without the wait, 11.5 of the 12.8 points of wrong stops at $L{=}8$ are on the pre-edit answer; with the wait $L{+}2{-}p$, none are at either length. A label-aware oracle that stops where the warm start's first run of eight correct readouts ends exits after 13.87 updates at $L{=}8$ (12.67 at $L{=}6$), 0.03 (0.07) updates before the recipe.

Table 40: **Reusing latent states after a prompt edit, with plain prefix caching as the baseline** (edit-ordered checkpoints; answer-changing edits at a uniformly drawn operand position p; stop on eight identical readouts counted after the stated wait). Compute: position-updates to the stop saved relative to recomputing from h_0, from per-p pooled *medians* averaged over p (brackets: per-checkpoint range). Lag and wrong stops: *means* over edits, with non-stops counted at lag 35. Rows 1–2 share their readouts by construction, as do rows 3–7.

	Positions			Compute saved (%)		Stopping lag		Wrong stops (%)	
Strategy	$<p$	$\geq p$ start at	Wait	L=8	L=6	L=8	L=6	L=8	L=6
Recompute	recomputed	h_0	none	0	0	18.2	16.0	0.52	0.00
Prefix caching	cache	h_0	none	45	44	18.2	16.0	0.52	0.00
Warm start	recomputed	cached state	$L{+}2{-}p$	25	22	13.9	12.7	0.43	0.00
Recipe	**cache**	**cached state**	$\boldsymbol{L{+}2{-}p}$	**56** [46–61]	**54** [43–60]	**13.9**	**12.7**	**0.43**	**0.00**
Recipe	cache	cached state	$L/2{+}2$	55	53	14.6	13.3	0.43	0.00
Recipe	cache	cached state	$L{+}1$	50	50	16.4	14.4	0.43	0.00
Recipe	cache	cached state	none	62	54	12.6	12.5	12.8	1.7

Statistics. Compute and the cost formula use medians: for each p we take the pooled median number of position-updates to the stop (lags up to 60), divide the average over p by recomputing's, and subtract the ratio from one. Stopping lag and wrong stops are means over edits, with stops up to lag 35 as in Appendix G.7. The product $(N{-}p_{\mathrm{c}})(N{-}p_{\mathrm{w}}{+}8)$ equals the pooled median in all 56 cells of four strategies and $8{+}6$ edit positions; per-checkpoint medians scatter around it (Figure 7a, band). The prefix cache holds the prefix's keys and values in every layer at every update of the unedited run. A warm start from update t_{cache} that exits at lag U reads them up to update $t_{\mathrm{cache}}{+}U$, so the saving assumes that the unedited run has been computed that far. Kept to three times the insertion time, the cache covers the recipe's exit in at least 99.2% of warm starts from the insertion-time cache, for each length and edit type; warm starts from three times the insertion time need it beyond that. If the unedited run stopped at the warm-start point, advancing its prefix would cost what reusing the prefix saves, and the saving would fall to that of warm starts alone (25% at L=8, 22% at L=6). The prefix read from the cache matches the recomputed one to float precision (E1, Appendix G.7). Cost counts position-updates only; it omits the cache's memory. Per checkpoint, the recipe's wrong-stop rate equals recomputing's in 35 of the 36 cells of 9 checkpoints, two lengths and two edit types, and is lower in the other; its mean stopping lag is shorter in all 36.

By edit position and edit type. Without the wait, the eight-repeat exit's wrong stops, frequent after edits at $p \leq 3$ (Appendix G.7, with $j = p{-}1$), stay at or below 1.4% of warm starts at $p \geq 4$ (L=8). The saving grows with p because both factors of the cost shrink. Counted to the settling onset s_8 (Appendix G.7), which needs labels, the recipe costs more than recomputing in 17% of warm starts at $p = 1$ and saves 63% averaged over p (62% at L=6). After answer-preserving edits ($p = 1, \ldots, L{-}1$) it saves 54% (51%); there the wait costs 6.5 extra updates on average at $p = 1$ in warm starts whose unwaited stop was already correct (L=8). The latency saving, 24% of recomputing's mean lag at L=8 (21% at L=6), ranges over 1.0–6.1 updates per checkpoint (0.4–5.2 at L=6).

Other stop rules. Keep eight identical readouts. A four-readout rule with the wait $L{+}3{-}p$, chosen on the edit-ordered checkpoints at L=8 (all edits), exits after 10.3 updates there but errs in 1.76% of edits on the other checkpoints, against 0.71% for the eight-readout rule with the wait $L{+}2{-}p$. Offline KL-threshold exits, which stop once the KL divergence between consecutive readout distributions falls below a threshold, need the wait too: on Experiment II's answer-changing transplants (all checkpoints, L=8, edits at the first two operands, so p=1 and $L{+}2{-}p = L{+}1$; thresholds 10^{-1} to 10^{-4}), they stop 52–90% wrongly without it and 1.3–7.9% with the wait $L{+}1$, against 0.46% for eight identical readouts after the same wait. The wait $L{+}2{-}p$ is not the shortest safe one: at both lengths, waits up to 3 updates shorter stop wrongly as rarely (Figure 7b shows L=8) and exit at most 0.1 updates sooner. Constant waits are as safe from 5 updates at L=8 (mean lag 14.2) and from 3 at L=6 (12.7), thresholds read off these data.

Outside the edit-ordered checkpoints. Over all checkpoints at $L{=}8$, the recipe saves 56% (41–61% per checkpoint) with 0.54% wrong stops, against 0.59% for recomputing. The cost formula holds only on the edit-ordered checkpoints; pooled over the others, recomputing's median exit comes before $N{+}8$ updates. On two of them, A_5 s2 and $(\mathbb{Z}_2)^6$ s1, some warm starts still read a wrong answer at lag 60 where recomputing reads the right one (up to 16 of 480 warm starts over both edit types, $(\mathbb{Z}_2)^6$ s1, $L{=}8$; none on the edit-ordered checkpoints), and the recipe's wrong stops exceed recomputing's by up to 6.25 points over lengths and edit types. Such errors show only against a reference run. A small edit sweep flags both checkpoints: on the edit-ordered checkpoints the error peak moves by -1.21 to -0.57 updates per position of the edit, and on the other seven, these two among them, by -0.39 to 0.25 ($L{=}8$; Table 36). In these data the recipe stops wrongly more often than recomputing only outside the edit-ordered checkpoints. For another model, run such a sweep and compare reused with recomputed exits on sample edits before relying on reuse.

Auditing with the continuation check. The continuation check (Table 2) compares a state's next $L{+}1$ readouts with a reference run's, here the untouched run; we did not evaluate it for reuse, where recomputing has not yet settled at lag $L{+}1$ (median settling onset 11 at $L{=}8$). On Experiment II's both-correct answer-preserving transplants, it flags 56.6% of states at $L{=}8$ (55.8% at $L{=}6$), and 99.6% of the flagged states (99.8%) read the right answer again at lag 35. In every task and length, at least 97.4% of the correctly decoding transplants that later err first do so by lag $L{+}1$. Of the states it passes, 1.72% (2.10%) err later, between lags $L{+}2$ and 35; per checkpoint, the horizon at which none does ranges over 5–13 updates at $L{=}8$ (4–14 at $L{=}6$). The check costs $2(L{+}1)$ updates per state, 18 at $L{=}8$ (14 at $L{=}6$), about recomputing's mean exit lag (18.2 at $L{=}8$), so it audits a method on sample states and cannot gate each state. A reference-free variant that only requires the state's own readout to stay unchanged for $L{+}1$ updates passes 1.67% of answer-changing transplants at $L{=}8$ when it skips the correctness check, all of them wrong.

G.9 Insertion time and displacement direction

Both analyses in this subsection are exploratory. Their scripts stated predictions before running (file times 10:49 and 11:11; first outputs 14:57 and 15:11), but the operational options that decide those predictions were compared only after the runs and were adopted by the authors after the runs (for the direction battery the authors accepted the existing pair selection after the run; ledger item "Direction-control pairs"). Because the adoption followed the runs, we report the pre-stated statistics descriptively.

Inserting after the recipient has settled. Could the loss come from two unfinished computations colliding at t_{80}? To test this, we insert the same answer-preserving transplants at update 40, when the recipient has nearly stopped moving (median one-update relative change 5.6×10^{-6}, against 4.2×10^{-2} at t_{80}; the donor's is about the same). We use $L{=}8$, the first 64 pairs of blocks 0–3 per checkpoint, and the pairs whose sham and transplant read the target at lag 0 at every insertion time; the sham itself errs in none of them (0.000) from update 40 on. The error of the same pairs is almost unchanged (Table 41): 0.591 after insertion at t_{80} and 0.580 after insertion at update 40, a ratio of 0.982 [0.968, 0.996] over all 828 pairs (90% interval; lags 1–8: 0.978 [0.963, 0.993]). Per checkpoint the ratio lies between 0.87 and 1.03. These errors are transient as well (0.0012 of pairs are wrong at lag 35). Reusing a settled state is thus no safer, and a collision of unfinished computations cannot by itself explain the loss. What persists after settling is that the transplanted operand states were computed for other tokens than the recipient's prompt supplies.

Direction, not only size. With each seam position's displacement norm matched to the donor's, a random direction inside the span of that position's native states errs far more often than an isotropic one (Table 42). The two rates are 0.45 and 0.02, a difference of 0.43 [0.42, 0.44], separated in 24/24 cells; the in-span rate reaches 0.82 [0.79, 0.86] of the donor's own rate (0.28 to 1.46 across cells). The native span is fitted per position by PCA of t_{80} states, cross-fitted by context, at the smallest rank that explains 95% of held-out variance. The sample is stratified (per cell, from all evaluation blocks, 48 pairs whose archived donor errs and up to 48 whose donor does not; A_5 s3 has only 34 and 41 non-erring pairs at $L{=}8$ and $L{=}6$), so the rates are not population rates, the pairs differ from those of Runs A–C, and only the displacement norm is matched, not the state norm. The projection of the donor displacement onto the span and a sibling donor's displacement are close to the donor's

Pairs	n	error, insertion at t_{80}	error, insertion at update 40	ratio [90% CI]
Pre-declared roster (9)	468	0.577	0.566	0.981 [0.968, 0.993]
Other checkpoints (7)	360	0.608	0.597	0.982 [0.955, 1.005]
Robust four	214	0.584	0.584	1.000 [1.000, 1.000]
A_5 (5 seeds)	251	0.558	0.546	0.979 [0.958, 0.993]
D_{31} (3)	155	0.677	0.671	0.990 [0.972, 1.000]
$\mathbb{Z}_{61}$ (4)	213	0.516	0.493	0.955 [0.920, 0.990]
$(\mathbb{Z}_2)^6$ (4)	209	0.641	0.641	1.000 [0.970, 1.031]
All 16	828	0.591	0.580	0.982 [0.968, 0.996]

Table 41: **Insertion after settling** [X], L=8, answer-preserving transplants, the same both-correct pairs at both insertion times, errors in lags 1–20. Intervals resample each context jointly across the seeds of its task (2,000 draws, 90%). The script's pre-stated band for "unchanged" was a ratio in $[0.75, 1.25]$; every unit lies inside it. In the robust four the two insertion times give the same number of erring pairs in every context cluster, so the interval is degenerate.

own by construction and are not independent evidence. The component outside the span still errs (0.10), so we do not claim that it is harmless. This battery tests whether direction matters. It is not a test of chaos.

Start (only seam positions changed, displacement norm per position matched to the donor's, unless stated)	error rate [95% CI]
Sham	0.01 [0.01, 0.02]
Full transplant (all positions)	0.51 [0.48, 0.53]
Donor displacement at the seam (the seam-only transplant)	0.55 [0.53, 0.57]
its projection onto the native span, renormalised	0.56 [0.54, 0.58]
its component outside the native span, renormalised	0.10 [0.09, 0.12]
Random direction inside the native span	0.45 [0.44, 0.47]
the same, drawn with the span's PCA variances	0.63 [0.62, 0.64]
Isotropic direction, radial part removed	0.02 [0.02, 0.03]
isotropic, radial part and state norm matched to the donor's	0.04 [0.04, 0.05]
Sibling edit's donor displacement at the seam	0.59 [0.57, 0.61]
Whole-state isotropic displacement of norm $\|v\|$	0.05 [0.05, 0.06]
Whole state pulled toward a random state of the pool's radius, distance $\|v\|$	0.19 [0.18, 0.20]
Donor direction, rescaled to the distance of a random pool state	0.99 [0.98, 0.99]

Table 42: **Direction battery** [X], mean over 24 cells (16 checkpoints at L=8, the eight A_5 and D_{31} checkpoints at L=6) of the share of pairs with any error in lags 1–20. Pairs are stratified by whether the archived donor errs, so the rates are not population rates. Intervals resample contexts within each cell (2,000 draws), with cells held fixed.

G.10 Tests of a transient-chaos reading: methods and complete results

Section 5 asks whether the late loss is the sensitive dependence reported for looped models (Lai et al., 2026). The object differs from theirs: they map convergence time from random initial latent states, whereas we follow the answer after a mid-computation transplant. Below we give methods and complete numbers for each tested signature. All results are exploratory and at L=8 unless stated. All rates are finite-time quantities along the sham trajectory from t_{80}, where the recipient itself is still settling. Most rows include all 16 checkpoints (the kicks, the fixed-frame exponents, the finite-window gains and the fate-map refinement); the co-moving peak and the error-time row are stated for the roster, and the boundary-entropy planes cover 9 checkpoints that differ from the roster (A_5 s0–s4, D_{31} s0, s1, s4 and $(\mathbb{Z}_2)^6$ s1; none in $\mathbb{Z}_{61}$). We read the negatives as a dissociation from Lai et al. (2026), not a refutation (Section 5); their scope is stated at the end of this subsection.

Distances are Euclidean norms of state differences over all positions and all 768 coordinates (7,680 dimensions at L=8), taken in the recurrent state before the exit blocks and their final normalisation. At length 8 the full transplant's distance to the sham stays below its lag-0 value at every lag 1–20 in 97.0% [95.7, 98.2] of 830 both-correct pairs.

Small perturbations. We add isotropic random kicks of size $\varepsilon\,\|v\|$ to the transplant (48 directions per pair, ε on a logarithmic grid) and ask whether the lags-1–8 readout sequence changes. At ε =

9.7×10^{-3} a kick changes it in 0.41% [0.19, 0.65] of kicks (mean over 830 both-correct pairs of all 16 checkpoints of the per-pair share); 16 pairs show any change at that size or below. Along the segment $h_S + s\,v$ we evaluate 97 points on $s \in [-0.25, 1.25]$ (step $1/64$) and refine each interval in which the label sequence of lags 1–8 changes into 17 parts ($1/1{,}088$) and again ($1/18{,}496$), at most 64 intervals per level. Only the first refinement lies at least 100 times above the numerical floor in almost all pairs. From the sham ($s = 0$) to the transplant ($s = 1$) the error indicator of lags 1–8 switches exactly once in 0.937 [0.914, 0.959] of erring pairs; this is an upper bound, because intervals whose two ends agree are never refined. The first switch lies at a median $s^* = 0.463$ of the donor displacement in erring pairs (0.77 over all sham-correct pairs, censored at 1.25). The answer is thus lost only after a finite, not an infinitesimal, displacement. The kicks test the size of a displacement, not its direction, and their power is limited: to reach a boundary at the median distance $s^*\,\|v\|$ within eight updates, a kick of size $9.7 \times 10^{-3}\,\|v\|$ must grow by a factor of about 50, which needs an exponent of about 0.48 per update if it lay along the growing direction and about 1.0 per update for an isotropic kick with one growing direction among 7,680 (our rough estimate, assuming the boundary lies at that distance along the grown direction). The largest 8-update rate that the power iteration (Appendix G.3) found, $\ln \sigma_1/8$ over the sampled untouched and transplant states, is 0.42 per update (rounded; an A_5 s4 transplant state; a lower bound on a sample), under the aligned requirement. The kick row therefore excludes only strong growth.

Co-moving exponents (propagator run). For 32 both-correct pairs per checkpoint (the first 16 that err and the first 16 that do not in lags 1–8), tangent vectors are carried 20 lags along the sham trajectory by Jacobian–vector products and renormalised every lag. The families are an isotropic random direction at position 1, the donor displacement restricted to position 1, and the whole donor direction. The co-moving (velocity-dependent) exponent $\lambda(v)$ (Deissler & Kaneko, 1987) is the OLS slope over lags $1, \ldots, \min(12, \lfloor 8/v \rfloor)$ (lags 1–6 for $v = 0$) of $\ln \|\delta_t[1 + \mathrm{round}(vt)]\|$, for $v \in \{0, 0.5, 0.75, 1, 1.25, 1.5\}$ positions per update. Because every tangent is carried along the sham trajectory, the split into erring and non-erring pairs changes only the contexts, not the trajectory along which growth is measured; exponents along the transplant's own trajectory, where a transient-chaos reading predicts growth, were not computed. The lag window shrinks as v grows, so the location of the peak partly reflects the window; a common window was not tried. Pooled over the roster (2,000 draws, contexts resampled jointly across the seeds of a task), $\lambda(v)$ is negative at every v and peaks at $v \in [0.7, 1.3]$ in 1.00 of draws for every family. At $v = 1$, $\lambda(1) = -0.087$ [-0.098, -0.077] for the isotropic direction, -0.073 [-0.096, -0.057] for the donor direction at position 1 and -0.135 [-0.146, -0.121] for the whole donor direction: a disturbance carried at about one position per update that decays weakly in the moving frame. In A_5 s4 the maximum lies at the edge of the grid ($v = 1.5$). Of the checkpoint-level maxima (16 checkpoints, three families, two grids), 4/96 have intervals above zero without correction for multiplicity. 1 of them is on the standard grid: A_5 s0, a roster checkpoint, at $v = 1.0$ (0.019 [0.005, 0.065], donor direction at position 1). 3 are on a finer grid added after running: A_5 s0 again, same family, at $v = 0.875$ (0.025 [0.013, 0.066]), inside the range reported above, and 2 in A_5 s4 at $v = 1.75$ (0.080 [0.026, 0.143] for the donor direction at position 1, 0.063 [0.007, 0.112] for the isotropic one), outside the range $[0, 1.5]$. Weak positive co-moving growth near one position per update is therefore open in A_5 s0; A_5 s4 shows weak growth only at $v = 1.75$, outside $[0, 1.5]$. Outside the roster the isotropic peak falls in $[0.7, 1.3]$ in only 0.05 of draws, so the speed there is not resolved by this grid. The script stated predictions before running; the statistics that decide them were proposed by the analysing agent before the run and adopted by the authors only after it, and no as-written copy was kept, so we report them as exploratory. $\lambda(0)$ is clearly negative in 8 of 9 roster checkpoints, and a whole-donor peak near $v = 1$ with its maximum near zero appears in 5 of 9. No fixed-frame exponent is positive in 16 of 16, over lags 1–20 only. Donor tangents lie at least 0.1 above isotropic ones in 0 of 16, and a sensitivity maximum moves at 0.8–1.2 positions per update in 8 of 9. A first-order certificate predicts which pairs err in at most 3 of 16 under any of 24 readings.

Fixed-frame exponents and finite-window amplification. Table 43 gives the fixed-frame exponents of all 16 checkpoints over lags 1–8, when errors occur, and over lags 1–20. Over lags 1–20, the median exponent of every measured family is negative in every checkpoint (-0.48 to -0.20 per update in the roster for the least-contracting family); these seven families are not the maximal direction (below). Over lags 1–8 the roster barely contracts (-0.20 to -0.02), and the intervals of 3 checkpoints (A_5 s3, s4 and D_{31} s1) reach zero, whereas the other seven contract clearly (-0.42 to -0.22). Growth over a finite window exists along some directions. We difference the 8-update map

at $\epsilon = 10^{-2}\,\|v\|$ along 33 orthonormalised directions ($\hat{v}$ and 32 random ones) and take the largest singular value σ_1 of the resulting $7{,}680 \times 33$ matrix, a lower bound on the largest singular value of the full Jacobian (64 pairs per checkpoint). It exceeds one in 17–39% of pairs in 4 checkpoints (A_5 s3, s4, D_{31} s4, $(\mathbb{Z}_2)^6$ s4), in 1.6–9.4% in 4 more and in none of the remaining 8. The map is differenced at the sham state at insertion, for all 64 pairs per checkpoint (no both-correct filter), and the gain was not split by whether the pair errs. Such transient amplification also occurs in non-normal systems without chaos (Hennequin et al., 2012), so this row does not discriminate. A later power iteration on the full Jacobian at 4 distinct sham states per checkpoint (Table 28; Appendix G.3) estimates the maximal growth from below: the 8-update map expands along its top direction at every such state in 12 of 16 checkpoints, all 9 roster checkpoints among them, and at half of them or more in 15, whereas the median estimate for the 32-update map is below one in 15 of 16. The exception, $(\mathbb{Z}_2)^6$ s4 outside the roster, keeps a median near 14.6 up to 32 updates. At the same untouched states the 33-direction bound is a median 12 times smaller than the power-iteration estimate (4.4–20.8; above one in 12 against 112 of 128 sampled pairs). Finite-window amplification along the maximal direction is thus common, including in checkpoints whose 33-direction bound stayed below one; as a lower bound, a 32-update estimate below one does not prove contraction.

Checkpoint	pairs/contexts	lags 1–8 [95% CI]	family	lags 1–20 [95% CI]	family
Pre-declared chain roster					
A_5 s0	32/19	−0.072 [−0.096, −0.053]	donor_full	−0.208 [−0.267, −0.158]	donor_p2
A_5 **s1**	32/22	−0.063 [−0.104, −0.032]	donor_p1	−0.374 [−0.388, −0.348]	donor_p2
A_5 **s3**	32/24	−0.024 [−0.044, 0.018]	donor_p1	−0.345 [−0.362, −0.312]	donor_full
A_5 **s4**	32/19	−0.030 [−0.089, 0.044]	donor_p1	−0.482 [−0.496, −0.456]	donor_full
D_{31} s0	32/17	−0.203 [−0.246, −0.120]	donor_p1	−0.396 [−0.417, −0.349]	donor_full
D_{31} s1	32/20	−0.029 [−0.074, 0.004]	donor_p1	−0.292 [−0.314, −0.270]	donor_p2
D_{31} **s4**	32/17	−0.044 [−0.104, −0.001]	donor_p1	−0.343 [−0.364, −0.301]	donor_p1
$\mathbb{Z}_{61}$ s0	32/18	−0.149 [−0.188, −0.084]	donor_p1	−0.362 [−0.373, −0.342]	donor_p2
$\mathbb{Z}_{61}$ s4	32/23	−0.156 [−0.175, −0.123]	donor_p2	−0.198 [−0.217, −0.170]	donor_full
Other checkpoints					
A_5 s2	32/22	−0.218 [−0.243, −0.197]	donor_p2	−0.225 [−0.238, −0.168]	donor_p1
$(\mathbb{Z}_2)^6$ s0	32/19	−0.282 [−0.313, −0.249]	donor_full	−0.266 [−0.292, −0.223]	donor_p1
$(\mathbb{Z}_2)^6$ s1	32/18	−0.408 [−0.442, −0.365]	donor_p4	−0.363 [−0.383, −0.305]	donor_p4
$(\mathbb{Z}_2)^6$ s3	32/19	−0.421 [−0.452, −0.396]	donor_p4	−0.361 [−0.422, −0.332]	donor_p4
$(\mathbb{Z}_2)^6$ s4	32/18	−0.397 [−0.438, −0.315]	donor_p1	−0.379 [−0.434, −0.343]	donor_full
$\mathbb{Z}_{61}$ s1	32/17	−0.336 [−0.363, −0.291]	donor_p1	−0.359 [−0.368, −0.343]	donor_p4
$\mathbb{Z}_{61}$ s3	32/21	−0.390 [−0.439, −0.341]	donor_p2	−0.347 [−0.370, −0.324]	donor_full

Table 43: **Fixed-frame growth exponents over lags 1–8 and 1–20** [X], L=8. Linearised along the sham trajectory, tangent vectors are propagated for each pair from seven direction families (the donor direction and random directions at positions 1, 2 and 4, and the whole-state donor direction). For each family we take the median over pairs of the least-squares slope of the log whole-state norm against lag (per update); the entry is the largest of the seven medians, i.e. the least-contracting family. Intervals resample contexts (2,000 draws). The robust four are in bold. Decision rules for this analysis were fixed after it ran; it is exploratory.

Error time. Chaotic amplification would let the error time scatter independently of where the edit sits. Instead it tracks the edited position (Table 36; slopes -1.21 to -0.57 updates per position in the roster at L=8), and E3 re-estimates this on new edits of the same contexts (Appendix G.7). The roster was selected on this property, so this row restates the selection criterion and is not independent evidence, and E3's re-estimate on the same contexts inherits the selection.

Fate maps: refinement. For a stratified sample of 206 both-correct pairs (16 checkpoints, 186 context clusters; the strata oversample pairs near the error boundary, so shares are not population rates), labels along $h_S + s\,v$, $s \in [-1, 2]$ in steps of $0.01\,\|v\|$, are refined 16-fold around every change. For 134 refined intervals, all on long lines of $(\mathbb{Z}_2)^6$ s1 and none for the error indicator, the refined points were not saved; they are excluded below. Only the first refinement (step $6.25 \times 10^{-4}\,\|v\|$) clears the threshold of 100 times the numerical floor, and only in 190/206 pairs (none in the rest). At that level every coarse switching interval of the lags-1–8 error indicator holds exactly one fine switch (286/286, in the 157 pairs with a coarse switch; the refinement ratio ρ, fine switches per coarse switching interval, is 1.000, its smooth value). The multi-valued outcomes gain a few fine switches at that level: 622/615 fine over coarse switches for the first-error lag ($\alpha = 1 - \log_{16}\rho$ in

[0.992, 0.999], excluding 1), 578/576 for the first wrong label and 1024/960 for the last label change (28/960 coarse switches split; 1133/1026 with the unsaved intervals). Counted over every refinement level whose step lies above the floor rather than at the admissible level only, 10/1497, 8/1361 and 36/2277 coarse switches split. Over $s \in [0, 2]$, 155 of the 206 pairs switch at the coarse level at least once, and the boundary is not a single threshold: 29/155 switching pairs switch more than once, so the script's expectation of exactly one switch in at least 90% of pairs fails: 126 of all 206 pairs switch exactly once (0.61 [0.54, 0.68]), counting as the rule does the 51 pairs that never switch, and 0.81 [0.75, 0.88] of the 155 that switch. "One switch" holds locally, per coarse interval, not globally. The script's two-level falsification rule cannot be applied as written, because 0/206 pairs have two admissible 16-fold levels. After the analysing agent had evaluated all options on the saved data, the authors chose (2026-09-25, about 17:00 JST) to apply the rule to a four-fold re-split of the saved level-1 points (post hoc; no new run). Then 190/206 pairs have at least two admissible levels, every O1 coarse switch at each of the 3 admissible levels holds one fine switch (627 switches; counted over every level whose step lies above the numerical floor, admissible or not, 0 of 1,584 switching intervals split), and the smallest exponent over outcomes and levels is $\alpha = 1 - \log_4 \rho = 0.953$ (O4), so the rule's threshold $\alpha < 0.85$ does not fire (0 levels). The positive control below shows that this threshold has no power at one 16-fold level; its power under the four-fold re-split was not estimated.

Positive control for the refinement. Could this refinement detect a fractal boundary at all? We regenerated 40 synthetic fractional-Brownian fields (Hurst exponent 0.8, boundary dimension 1.2, the construction of the synthetic references of the state-plane analysis below). We generated them at the run's resolution, 4,801 fine points per 300 coarse intervals, thresholded them at their median, and took rows and columns every 16 pixels as transects. Each replicate draws 286 coarse switching intervals from transects whose number of coarse switches matches that of the 157 real pairs. It then applies the same one-level 16-fold refinement and the same statistic, with an interval that resamples transects. The ratio ρ is 1.18 (1.11–1.26 across replicates), and its interval excludes 1 in 1000 of 1000 replicates. The replicates are drawn from the same 40 fields (a median of 39 per replicate), so they are not independent, and the Clopper–Pearson lower bound of 0.996 overstates the precision of the detection rate; an interval that resamples whole fields instead of transects also excludes 1 in 1000 of 1000. A median of 24 (15–33 in 95% of replicates) of the 286 coarse switches split, and 0 replicates leave all of them unsplit, as the model does. The result is the same without matching (1000 of 1000), and the 20 original references give $\rho = 1.26$. At dimension 1.5 detection is 500 of 500, and at 1.8 (unmatched, because no transect has as few switches as the real ones) 500 of 500. A smooth negative control (Gaussian fields of long correlation length, unmatched) fires in 0 of 500 replicates under every criterion; all 12190 of its coarse switches are unsplit. The scripts' own fractal threshold, $\alpha < 0.85$ (read as $\alpha = 1 - \log_{16} \rho$; the docstring's reading on the median number of fine switches, above one, fires in none either), would not have fired at dimension 1.2 (0 of 1000; α is 0.94, 0.92–0.96), although it fires at dimension 1.5 in 499 of 500. The evidence therefore rests on unsplit coarse switches, not on that threshold. The control excludes fractional-Brownian-type boundaries of dimension 1.2 or more that are self-similar from the transect length down to $6.25 \times 10^{-4} \, \|v\|$; dimensions between 1 and 1.2 were not tested. A boundary that is rough only below this scale is untested, and a boundary crossing a coarse interval an even number of times is invisible here and in the control alike. The control was run post hoc; its detection criterion (the interval of ρ excluding 1, with at least 80% detection) was fixed in the agent-written analysis plan before it ran; the authors adopted it after the run.

State planes. For 31 erring pairs (12 context clusters) a 128×128 grid spans v ($a \in [-1, 2]$) and a random direction $u \perp v$ with $\|u\| = \|v\|$ ($b \in [-1.5, 1.5]$). A pixel is $0.024 \, \|v\|$ and the plotted outcome is the lags-1–8 error indicator. The second axis is random in every plane and moves the readout little, so islands could be missed. Box counting over 2–16 pixels gives 1.020 [0.998, 1.047]. On synthetic 128×128 maps (20 each) the same estimator returns 1.00–1.01 for straight and tilted lines, 0.98 and 1.04 for contours of Gaussian random fields with correlation lengths 16 and 8 pixels, and 1.14, 1.36 and 1.55 for fractional-Brownian boundaries of true dimension 1.2, 1.5 and 1.8. The estimator is biased downward: a pooled true dimension of 1.2 would have given about 1.14, above the observed interval, but smaller excesses would not be detected. 10 of the 31 planes exceed the largest value that any smooth reference reached on at least one of five estimators (box counting, two uncertainty-exponent and two length-scaling estimators); 8 exceed it on two or more and 1 on all five. How often a new smooth boundary would exceed that maximum was not estimated, so this count is not calibrated. Zoomed planes spanned by v and a random orthogonal

direction (half-widths $0.1\,\|v\|$ and $0.01\,\|v\|$ around the first switch; 72 planes in nine checkpoints) have one connected component per class and no islands, although only 8/36 of the finest planes are admissible. At half-width $\|v\|$, 10/36 planes have a class in more than one component.

Boundary entropy. Lai et al. (2026) summarise convergence-time maps by the basin entropy S_b and the boundary entropy S_{bb}: the mean Shannon entropy of the values in non-overlapping boxes, over all boxes or over boxes holding more than one value. In the basin-entropy literature on which they build, $S_{bb} > \ln 2$ is a sufficient condition for a fractal boundary (Daza et al., 2016). On the 108 planes of 9 checkpoints (three zooms), we take as outcome the last label change in lags 1–20, the closest analogue of their convergence time, and also the first wrong label. With boxes of 4, 5 and 8 pixels, S_{bb} stays below $\ln 2 = 0.693$ in every plane: at most 0.676 for the last change and 0.632 for the first wrong label (0 and 0 of 108; 0 of the 80 admissible planes). This criterion is one-sided and was not predicted. Failing it shows only that its sufficient condition was not met; it does not show smoothness. Quantised outcomes can also come close to it without roughness: synthetic smooth three-valued bands one pixel wide exceed $\ln 2$ in 13–57% of planes, depending on box size. Bands at least three pixels wide exceed it in at most 13%, and bands at least six pixels wide in at most 0.4%. The largest observed value comes from a one-pixel-thick sliver of a third value, broken into small islands. At the finest zoom the two outcomes give identical values (8/36 planes admissible).

Transient wrong labels. In the transient-chaos reading of Lai et al. (2026), slow passages decode to nearly correct attempts. Our first wrong readout labels equal a one-operand hybrid product (H_1 or H_2) in at most 0.07 and 0.04 of erring pairs per roster checkpoint at L=8 (0.19 in A_5 s0 and 0.07 in A_5 s1 at L=6). These values lie above the chance of a particular wrong label (1.6–1.7%). A post hoc one-sided exact binomial test against that uniform choice among the $|G| - 1$ wrong labels, Bonferroni-corrected over the 36 roster cells (nine checkpoints, two lengths, H_1 and H_2), finds a share above chance in 1 cell (A_5 s0, H_1, L=6: 11 times chance). The uniform null is weak, since any structure among wrong labels raises hybrid frequencies. A one-operand hybrid is also what a carried stale operand would produce, so this row does not discriminate between the readings.

ε-scan slopes and the Countdown probe. The log–log slopes of ε-scans are not uncertainty exponents in this design. For one flat boundary at distance d seen by isotropic random directions in $D = 7{,}680$ dimensions, the fraction of ε-perturbations that cross it is $\Phi_c(d\sqrt{D}/\varepsilon)$. Its local slope falls from 4.97 at a crossing fraction of 0.02 to 0.61 at 0.3, so even a smooth boundary gives any slope in that range. The Countdown probe uses a separate Countdown Parcae model (Prairie et al., 2026) (12 contexts, prompt lengths 14–19). At update 8, 27 or 70 one position receives a random kick of 10^{-2} of its norm (one direction per position). Arrival is the first lag at which the last position's divergence reaches 10% of its maximum over lags 1–80. The kick arrives in one update in 492/492 cases. The slope regresses the lag of that divergence's peak on the kicked position's distance (start position excluded): 0.007 [−0.009, 0.024] per position. A small random kick differs from a finite structured edit, we did not test this model for chaos, and its weights are the ones shown in Figure 6 (same SHA-256 prefix `517c3bd9`).

The main text's Table 1 in full. Table 44 gives every number behind the six rows of Table 1. For transplants, our reading of transient chaos (Lai et al., 2026) predicts rows 1, 2 and 4–6; only rows 2 and 5 could discriminate it from a carried difference. Row 5 is absent for errors, and row 2 is negative pooled over the roster (open in A_5 s0 and s4, without correction for multiplicity). Row 1's kicks are too small for row 3's growth, row 4 is circular because the roster was selected on it, and row 6 exceeds chance in 1 of 36 cells, where a stale operand would give the same labels. Two facts fit both readings: errors are transient (Section 4.3), and finite-window amplification (row 3) also occurs in non-normal transport (Hennequin et al., 2012). Our object (the answer's fate after a transplant) differs from theirs (convergence time), and a group product needs no trial and error, so we read a dissociation, not a refutation.

Refinement: power, fold scale and the scripted criteria. Refinement detects every-scale roughness. On 40 synthetic fractional-Brownian boundaries of dimension 1.2, matched in resolution and coarse switches, it splits a median 24 of the 286 coarse switches (15–33 in 95% of replicates), and ρ's interval excludes 1 in 1000/1000 replicates (the replicates share these fields; the criterion is post hoc; smooth control: 0/500). It cannot see row 3's growth: at the largest measured rate, an 8-update

	Transient chaos predicts	What we measured, and found	Reading
1	Small differences change the outcome	16 checkpoints: 0.41% [0.19, 0.65] of isotropic kicks of $9.7 \times 10^{-3}\,\|v\|$ to the transplant change its lags-1–8 readouts (roster 0.39%); reaching the median error onset $s^* = 0.463$ along $h_S + s\,v$ in 8 updates needs $\approx$1.0/update ($\geq$0.48 if aligned; largest measured 0.42)	Not seen; kicks too small for row 3's growth
2	Growth in some moving frame (necessary)	Roster pooled, frames 0–1.5 positions/update: negative in every frame and resample, largest near 1 ($\lambda(1) = -0.087$ [-0.098, -0.077] isotropic, -0.135 [-0.146, -0.121] donor). Above 0 (uncorrected; 48 per grid): A_5 s0 at speed 1.0, 0.019 [0.005, 0.065], and 0.875 on a post hoc grid to 2, 0.025 [0.013, 0.066]; A_5 s4 at 1.75 (past its edge maxima), 0.080 [0.026, 0.143], 0.063 [0.007, 0.112]. Fixed frame, lags 1–20: 16/16 < 0	Not seen pooled; A_5 s0, s4 open; peak speed window-dependent
3	Amplification over a finite window	16 checkpoints, 4 untouched states each (first 8 of 16 scripted pairs, 10 of 15 steps), power iteration: σ_1 of the 8-update map > 1 at half or more in 15 checkpoints (at all in 12, the roster included); of the 32-update map < 1 at 34 of 36 roster states (not 2 in A_5 s0; repeat starts within 4.2%), median < 1 in 15 ($\leq$0.66), not $(\mathbb{Z}_2)^6$ s4 (14.6)	Seen at 8 updates; not at 32 (lower bounds); not discriminating
4	Error times scatter, whatever the edit	Roster: error peaks shift -1.21 to -0.57 updates per edit position (-1.20 to -0.40 at L=6); E3, new edit elements on the same contexts: in the predicted $[-1.5, -0.5]$ in 9 of 9 at L=8 and 9 at L=6	Not evidence: roster chosen on it
5	Refinement adds switches	16 checkpoints, stratified: at the one 16-fold level above precision, the 286 coarse error switches (in 157 of 206 pairs, $s \in [-1, 2]$; roster 127/127) each hold one fine switch ($\rho = 1.000$; dimension-1.2 control 1.18). Timing outcomes split a few: first-error lag $\rho = 1.01$ [1.003, 1.022], last change 1.07 [1.04, 1.10]	Absent for errors (power: every-scale roughness); timing undecided
6	Wrong outputs are near-misses	Roster, seam-only: first wrong labels are one-operand hybrids $(a_1 k)a_2 \cdots a_L$ or $a_1(k^{-1}a_2)\cdots a_L$ in per-checkpoint maxima of 0.07 and 0.04 (0.19, 0.07 at L=6); above chance ($\leq$0.017; binomial, Bonferroni) in 1 of 36 cells (A_5 s0, L=6: 11$\times$)	Rare (low power); also against a stale operand

Table 44: **Signatures of transient chaos, tested: complete version of Table 1** [X]. Length 8 unless stated; roster: the 9 edit-ordered checkpoints; both-correct pairs except row 3 and E3; v: transplant minus sham state h_S; $\lambda(c)$: growth rate at frame speed c, and σ_1 (a lower bound), around the untouched run; ρ: fine switches per coarse switch (1 if smooth); 95% intervals. Stated in scripts before their runs: E3 and parts of rows 2, 3 and 5; the rest post hoc. Methods in the paragraphs above.

readout folds no finer than about $0.016\,\|v\|$ (the arithmetic of row 1), above the $0.01\,\|v\|$ grid. The scripts' threshold ($1 - \log_{16}\rho < 0.85$) needs two admissible levels, which no pair has, and misses dimension 1.2 (0/1000); a four-fold re-split, adopted afterwards, splits 0 of 627 admissible error switches (its power is not calibrated). The boundary is locally one switch, not one threshold: one switch on $s \in [0, 2]$ in 0.61 of all 206 pairs and 0.81 of the 155 that switch, against the scripts' expectation of at least 90%. The last label change, closest to their convergence time, meets the control's criterion (28/960 split), but one level cannot tell nested level sets of an integer time from roughness, so it is undecided.

Scope of the chaos tests. We claim the refinement result (row 5) for all 16 checkpoints and the other negatives only for the edit-ordered roster, pooled for row 2 (off the roster, error time tracks the edit weakly if at all: slopes -0.39 to 0.25 at L=8); for growth around the untouched run, not along the transplant's own trajectory, where a transient-chaos reading places it (we computed no fast Lyapunov indicators and have no positive control in a real network); and for roughness of dimension at least 1.2 above $6.25 \times 10^{-4}\,\|v\|$. Signs that lean towards chaos sit in A_5 s0 (rows 2, 3 and 6) and, off the roster, in $(\mathbb{Z}_2)^6$ s4 (32-update $\sigma_1 \gg 1$; error delays like the roster's) and $(\mathbb{Z}_2)^6$ s1 (15 of 3,767 erring L=8 continuations stay wrong over lags 20–35, 14 of them in this checkpoint; their later return untested).

Checks before attributing late errors to chaos. Measure moving-frame growth along the intervened trajectory, not only along the untouched run; estimate maximal growth by power iteration; check whether error time tracks the edit; and calibrate boundary refinement with a positive control of known dimension.

G.11 PROPOSITION 2, ITS PROOF AND THE DELAY-LINE MEASUREMENTS

Proposition 2 (Causal recurrences as delay lines). *Let $\Phi_t = J_{t-1}\cdots J_0$ for one-update Jacobians J_t along the untouched run on positions $0,\dots,L{+}1$; block $J_t[i,j]$ (position j to i) vanishes for $j > i$. Assume $\|J_{t+m-1}[i,i]\cdots J_t[i,i]\| \le K\alpha^m$ (operator norms) with $0 < \alpha < 1$ for all i, t and $m \ge 0$ (so $K \ge 1$), and let $\nu \ge \|J_t - D_t\|$ for all t, with D_t the block-diagonal part of J_t. (i) $\|\Phi_t\| \le \sum_{q=0}^{\min(t,L+1)} \binom{t}{q} K^{q+1}\nu^q\alpha^{t-q}$: growth over a finite window is possible, yet every Lyapunov exponent is at most $\ln\alpha < 0$. (ii) If $J_t[i,j] = 0$ for $i - j \ge 2$, then $\Phi_t[i,j] = 0$ for $t < i - j$ and $\Phi_{i-j}[i,j] = J_{i-j-1}[i,i{-}1]\cdots J_0[j{+}1,j]$: to first order, a displacement at j has no effect at i before $i - j$ updates, and its first effect there passes through each position between. (iii) If positions after M read earlier ones only via h_M, the readout reads only them, the untouched run is constant in time on them and on M, and the displacement leaves them unchanged at insertion, then overwriting h_M from insertion on by its untouched value plus the displaced run's difference at M from k updates earlier (zero for $t{<}k$) delays their trajectory and every readout error by exactly k.*

Setting. The state $h = (h_0,\dots,h_{L+1})$ stacks the position vectors; an update is $h(t{+}1) = F(h(t))$, and $h^{\mathrm{S}}(t)$ is the untouched run. The blocks $J_t[i,j] = \partial F_i/\partial h_j$ of $J_t = DF(h^{\mathrm{S}}(t))$ vanish for $j > i$ (causal attention). Norms are Euclidean on the stacked state and operator norms on maps, so a block-diagonal map has the largest norm of its blocks. Write $J_t = D_t + N_t$ (block diagonal plus strictly lower), so $\nu \ge \sup_t \|N_t\|$. The stability bound is assumed for all i, $t \ge 0$ and $m \ge 0$; $m = 0$ gives $K \ge 1$. A spectral radius below one at each t would not suffice, because products of time-varying matrices can grow even so.

(i). Let $V_{\ge j}$ be the states supported on positions $\ge j$. Each D_s maps $V_{\ge j}$ into itself and each N_s maps it into $V_{\ge j+1}$. Expand $\Phi_t = (D_{t-1} + N_{t-1})\cdots(D_0 + N_0)$ into 2^t words. A word with q factors N maps every state into $V_{\ge q}$, and $V_{\ge L+2} = \{0\}$, so words with $q \ge L{+}2$ vanish. Otherwise its factors D form at most $q{+}1$ runs of consecutive times; a run $D_{s+m-1}\cdots D_s$ is block diagonal with blocks $J_{s+m-1}[i,i]\cdots J_s[i,i]$, so its norm is at most $K\alpha^m$. The run lengths sum to $t - q$ and $K \ge 1$, so the word has norm at most $K^{q+1}\nu^q\alpha^{t-q}$; summing over the $\binom{t}{q}$ words with q factors N gives the bound. It equals $K\alpha^t \sum_{q\le\min(t,L+1)} \binom{t}{q}(K\nu/\alpha)^q \le K\alpha^t P(t)$ for a polynomial P of degree $L{+}1$. Hence $\limsup_t t^{-1}\ln\|\Phi_t\| \le \ln\alpha < 0$, so every Lyapunov exponent of the linearised recurrence is negative, while the bound can exceed one at small t.

(ii). Block multiplication gives $\Phi_t[i,j] = \sum_{p_1,\dots,p_{t-1}} J_{t-1}[i,p_{t-1}]\cdots J_0[p_1,j]$. Under the band hypothesis of (ii), a nonzero summand is a path $j = p_0 \le \cdots \le p_t = i$ with steps 0 or 1, which needs $i - j$ steps of 1. None exists for $t < i - j$; for $t = i - j$ the only one is $p_s = j + s$, which gives the stated product. To first order, a displacement δ at position j changes position i at lag t by $\Phi_t[i,j]\delta$. Without the band hypothesis, split $J_t = J_t^0 + R_t$, with R_t the blocks with $i - j \ge 2$, and let $\|A\|_{\mathrm{blk}} = \max_i \sum_j \|A[i,j]\|$, which is submultiplicative and bounds every block. Expanding over the times at which R is chosen, the term without R vanishes in block (i,j) for $t < i - j$, so for such t, $\|\Phi_t[i,j]\| \le (a_0 + \varepsilon)^t - a_0^t$ with $a_0 = \sup_t \|J_t^0\|_{\mathrm{blk}}$ and $\varepsilon = \sup_t \|R_t\|_{\mathrm{blk}}$. The bound is at most $t\varepsilon(a_0 + \varepsilon)^{t-1}$, whose factor on ε can be large when $a_0 > 1$, so we measure the early response directly.

(iii). Let $\mathcal{D} = \{M{+}1,\dots,L{+}1\}$. The hypotheses are (C1) $F_{\mathcal{D}}(h) = G(h_M, h_{\mathcal{D}})$ for some map G; (C2) the readout is a function of $h_{\mathcal{D}}$; (C3) $h_M^{\mathrm{S}}(t) = \bar{h}_M$ and $h_{\mathcal{D}}^{\mathrm{S}}(t) = \bar{h}_{\mathcal{D}}$ for all $t \ge 0$, so $G(\bar{h}_M, \bar{h}_{\mathcal{D}}) = \bar{h}_{\mathcal{D}}$; (C4) the displaced run h^{P} has $h_{\mathcal{D}}^{\mathrm{P}}(0) = \bar{h}_{\mathcal{D}}$. The delayed run starts at $h^{\mathrm{P}}(0)$ and follows F, except that position M holds $r(t) = h_M^{\mathrm{S}}(t) + h_M^{\mathrm{P}}(t-k) - h_M^{\mathrm{S}}(t-k)$ for $t \ge k$ and $h_M^{\mathrm{S}}(t)$ for $t < k$; by (C3), these equal $h_M^{\mathrm{P}}(t-k)$ and $\bar{h}_M$. By (C1), $h_{\mathcal{D}}^{\Delta}(t{+}1) = G(r(t), h_{\mathcal{D}}^{\Delta}(t))$, so (C3) and (C4) give $h_{\mathcal{D}}^{\Delta}(t) = \bar{h}_{\mathcal{D}}$ for $t \le k$; in particular $h_{\mathcal{D}}^{\Delta}(k) = h_{\mathcal{D}}^{\mathrm{P}}(0)$. If $h_{\mathcal{D}}^{\Delta}(t) = h_{\mathcal{D}}^{\mathrm{P}}(t-k)$ for some $t \ge k$, then $h_{\mathcal{D}}^{\Delta}(t{+}1) = G(h_M^{\mathrm{P}}(t-k), h_{\mathcal{D}}^{\mathrm{P}}(t-k)) = h_{\mathcal{D}}^{\mathrm{P}}(t{+}1{-}k)$ by (C1) for the displaced run. By induction and (C2), the delayed run reads at lag $t \ge k$ what the displaced run reads at lag $t - k$, and before that what the untouched run reads, which is correct in both-correct pairs. Within an observation window of lags $1,\dots,W$, an error of the displaced run at lag $\tau \le W - k$ therefore reappears at $\tau + k$, and among pairs that err in both runs every first-error shift, hence their median, equals k. No linearisation is used. The delay of Section 4.2 is this construction with the operand-only transplant as the displaced run, $M = L/2{+}1$, $k \in \{2, 4\}$ and $W = 20$.

Scope. The proposition does not claim that trained checkpoints satisfy the stability bound, the band hypothesis or (C1)–(C3): later positions can attend to every earlier one, and insertion need not be at rest. The operand-only transplant changes only positions 1 and 2, and $M \geq 4$, so $h^{\mathrm{P}}_{\mathcal{D}}(0) = h^{\mathrm{S}}_{\mathcal{D}}(0)$ by construction, which is (C4) whenever (C3) holds. It gives sufficient conditions for delayed arrival and exact delays; the delay-line paragraph of Section 5 measures both. Per pair, 80–85% of the pairs that err in both runs shift by $k \pm 1$ and 12–15% not at all (Appendix G.3), so a median shift of exactly k agrees with (iii) without verifying (C1)–(C3). The proposition does not give the extra update in the wait $L{+}2{-}p$ of Section 6: (ii) concerns a displacement of the state, whereas a prompt edit changes an input.

An example that meets every hypothesis. Give each position $0, \ldots, L{+}1$ a scalar state, let one update be $h \mapsto \alpha S h$ with $0 < \alpha < 1$, $(Sh)_0 = 0$ and $(Sh)_i = h_{i-1}$, and let the logits be $(1, h_{L+1})$, the target first. No update enlarges a distance, and each multiplies it by at most α (the shift drops the last entry); every eigenvalue is zero, so there is no chaos. The stability bound (with $K = 1$ and $\nu = \alpha$) and the band hypothesis hold, and so do (C1)–(C4) for each M with $p < M \leq L$, for the displacement at position p below. The untouched run $h = 0$ reads the target at every lag. A displacement of size b at position $p \leq L$ with $b > \alpha^{-(L+1-p)}$ leaves the readout unchanged at insertion, errs at lag $L{+}1{-}p$ only, and equals the untouched run from the next lag on. For $p < L$, holding any position strictly between p and $L{+}1$ at the untouched run removes the error, and the delay of (iii) moves it to lag $L{+}1{-}p{+}k$. Late errors, their removal by an intermediate clamp and their exact delay therefore need no chaos; the example shows what is possible, not how the trained checkpoints compute.

Measured quantities (the propagator pairs of Appendix G.10: length 8, all 16 checkpoints, 32 both-correct answer-preserving pairs each, half of them erring in lags 1–8). For lags $t = 1, \ldots, 16$, g_t is the gradient of the untouched run's margin at lag t (target minus that lag's runner-up) with respect to the state at insertion, and v is the transplant's difference there. The response to positions P is $\sum_{i \in P} g_t[i]^\top v[i]$; a change of coordinates $v[i] \mapsto A_i v[i]$ maps $g_t[i] \mapsto A_i^{-\top} g_t[i]$ and leaves it unchanged. A pair's early/peak ratio is the largest absolute response to $P = \{1, 2\}$ over lags 1–3 divided by that over lags 1–16 (median over pairs). The height at distance d is $\|g_d[L{+}1{-}d]\|$, the largest response to a unit displacement there; per pair we fit its logarithm against $d = 1, \ldots, 8$ by least squares and report the median slope s as the change $e^s - 1$ per unit of d; the height counts as unchanged when $|s| \leq 0.03$. The 6 within it are A_5 s1, s3, s4 and D_{31} s0, s1, s4; A_5 s0, $\mathbb{Z}_{61}$ s0 and s4 fall outside. The early/peak ratio is 0.01–0.36 in the edit-ordered checkpoints (lowest A_5 s3, highest $\mathbb{Z}_{61}$ s4) and 0.71–1.00 in the other 7 (lowest A_5 s2). The count of 6 holds for any tolerance from 0.028 to 0.099; with each height multiplied by the norm of that position's state it is 5, as D_{31} s0 ends at $s = -0.030$. $\lambda(v)$, defined in Appendix G.10, follows the norm of a random displacement, whereas the height follows the direction the readout reads most, so a negative $\lambda(1)$ and an unchanged height are compatible.

A pre-fixed test of the band hypothesis. We tested whether, in trained checkpoints, the coupling that carries the errors is confined to nearest-neighbour blocks, as (ii) assumes. Along the untouched run, each arm iterates the nonlinear update of an operand-only transplant but passes its difference only through blocks (i, j) with $i - j$ in a kept set, all other blocks reading the untouched run's state: *band* keeps $\{0, 1\}$, *nohop* $\{0, 2, 3, \ldots\}$, *band2* $\{0, 2\}$ and *diag* $\{0\}$. R_X is the share of erring pairs (a wrong label in lags 1–20) that still err under arm X, and Q_X the share of non-erring pairs that err under it (length 8; 32 erring pairs per checkpoint). The rule, fixed before the run, scores the 4 most clearly edit-ordered checkpoints (A_5 s1, s3, s4, D_{31} s4) and contrasts six of the other seven (A_5 s2, $(\mathbb{Z}_2)^6$ s0, s1, s3, $\mathbb{Z}_{61}$ s1, s3). It supports the hypothesis if, in at least three of the four, $R_{\mathrm{band}} \geq 0.8$, $Q_{\mathrm{band}} \leq 0.1$, the median first-error shift under *band* is at most one update in size, $\max(R_{\mathrm{nohop}}, R_{\mathrm{band2}}) \leq 0.2$ and $R_{\mathrm{diag}} \leq 0.2$, and in at least four of the six $R_{\mathrm{band}} \leq 0.5$ or $R_{\mathrm{diag}} \geq 0.5$. It rejects it if at least two of the four have $R_{\mathrm{band}} < 0.5$, or at least two have $\max(R_{\mathrm{nohop}}, R_{\mathrm{band2}}) \geq 0.5$, or in at least two the share of the answer position's linear response kept under the band restriction and R_{band} disagree (one at least 0.8, the other below 0.5). The verdict is *reject*: R_{band} is 0.47 in A_5 s1 and 0.38 in A_5 s4, and only A_5 s3 meets the support clause. Descriptively, in the four, removing the nearest-neighbour blocks removes almost all errors (R_{nohop}, R_{band2}, $R_{\mathrm{diag}} \leq 0.031$), while keeping only them reproduces 0.38–0.91 of them: there, nearest-neighbour coupling is needed, and it suffices ($R_{\mathrm{band}} \geq 0.8$) only in A_5 s3. In 5 of the six others, longer-range coupling alone reproduces 0.41–1.00 of the errors (R_{nohop}). The main text therefore

makes no claim about the band structure of the trained checkpoints' Jacobians, and Proposition 2 is a conditional result.

G.12 HYPOTHESES WE TESTED AND DID NOT SUPPORT

Before Runs B and C, we tested local and auxiliary accounts of the failure on data that already existed; the rules were fixed in the analysis plan before each test was computed (self-reported). The seam-forcing, pair-timing and co-moving tests use saved single-precision states of 106 both-correct pairs at $L = 8$ (4–8 per checkpoint). The charge, $(\mathbb{Z}_2)^6$ and unsettledness tests use Experiment II's saved labels and logits (both lengths where stated), and the Countdown row uses the Countdown model. Table 45 lists each account, its rule and the outcome. Later readings of the effective map and the clamps are in Table 46; their tests were written after the data they use existed, except where a pre-declared prediction is named. The insertion-time control is in Tables 41 and 45 (collision row). Several of these tests have little power, so "not supported" does not mean "refuted" unless the table says so.

Account	Test and rule	Result	Status
Local seam forcing: the transplant's one-update excess motion is concentrated at the seam	Rejected if, in ≥ 5 of the 9 roster checkpoints, the seam's share of the lag-0-to-1 excess is at most its share of the displacement, or the downstream excess is $\geq 5\times$ the sham's own step	met in 6 of 9	rejected
Pair-level timing from the seam: first-error lag $= L + 1 - s^*$ at one position per update	Rejected if not better than a per-task constant for A_5 and D_{31} (median absolute error)	1.0 vs 1.0 over 29 erring pairs; the forecast is 7 or 8 in 28 of them	rejected; little power for the speed
Neutral co-moving transport	Largest growth exponent over frames moving with the disturbance in $[-0.25, +0.05]$ with upper bound < 0.10, in ≥ 14 of 16 checkpoints; "weak convective instability" if its lower bound exceeds $+0.05$ in ≥ 3	neutral in 8 (transplant) and 7 (small perturbation near the donor) of 16; of the other transplant checkpoints, 3 decay faster and 5 have intervals too wide to decide; unstable in 0	prediction failed; instability not seen; the propagator run finds weak decay in the moving frame (Appendix G.10)
Gauge charge: the conjugated edit decides which wrong label appears	Sibling pairs whose edits carry the same charge agree on the wrong label more often, by ≥ 0.10 with interval above 0 at both lengths; the error rate is monotone in the edit's rotation angle in ≥ 6 of 8 cells	gain $+0.028$ $[-0.15, 0.21]$ (L=8), -0.061 $[-0.22, 0.11]$ (L=6); angle law strict in 4 of 8	rejected by rule; low power; the angle criterion fires in most resamples even when the observed rates are true
One law for transplant errors and Countdown partitions (escape from a slow region)	Both states of the Countdown slice lie in the slow-escape region (last switch > 280)	last switch 255 and 39; predicted fine-scale slope drift (≤ -0.2) was $+0.006$	rejected (one pair, Countdown only)
Nonlinear switch in $(\mathbb{Z}_2)^6$	In ≥ 3 of 4 seeds, at both lengths, the minimum-margin distribution is bimodal (valley-depth ratio ≤ 0.5 and dip test $p < 0.01$)	0 of 4 seeds; smallest ratio 0.517	prediction failed
Collision of unfinished computations at t_{80}	Recipient unsettledness predicts errors with AUC ≥ 0.65 in ≥ 12 of 16 checkpoints	0 of 16 (largest 0.649); the rejection rule did not fire. Later, inserting at update 40, after the recipient settles, leaves the error unchanged (ratio 0.982, Table 41)	not supported
Chaotic saddle with a fractal fate boundary	As for neutral transport: positive co-moving growth in ≥ 3 checkpoints	0 of 16; propagator run: 4/96 checkpoint-level intervals above 0, uncorrected; refinement splits none of the 286 coarse switches where a dimension-1.2 control splits a median 24 (Appendix G.10)	not detected at a resolution whose power we established
Local message passing on prefix products	A downstream-only transplant has the recipient's tokens and prefix product at every transplanted position, so it should not err	errs in A_5 s0 (0.67), D_{31} s0 (0.37), $\mathbb{Z}_{61}$ s0 (0.47); this does not test what the seam itself carries	rejected in this form

Table 45: **Accounts tested and not supported, first round** [X]. The first eight rows are CPU tests on data that existed before they were run (rules fixed in the analysis plan beforehand, self-reported): saved states of 106 pairs at L=8 for the seam, timing, co-moving and saddle rows, Experiment II labels and logits for the charge, $(\mathbb{Z}_2)^6$ and unsettledness rows, and the Countdown model for its row. The last row is a check against Run A made after it was run; the collision and saddle rows add later evidence. Two implementations, a reconciliation and two verifications were run for every CPU test, all by AI agents.

Reading	Test	Result	Status
Exact inference over answers: before completion the answer position holds a near-uniform distribution over answers	EM5 (b), pre-declared: median normalised entropy of the native effective map ≥ 0.5 before it reads the answer, in ≥ 7 of 9. Exact inference from incomplete operands gives a uniform distribution over answers (entropy 1). A correct answer, once reached, should be kept	(b) in 2 of 9 at both lengths; median entropy 0.22 (L=8) and 0.19 (L=6) before completion (post hoc pooled probe); later errors in all 16 checkpoints (0.25–0.92 of pairs, Run A)	not supported (probe only; the readout itself untested)
The update is a running computation of prefix products	EM5 (a), pre-declared: the effective map's label is a prefix product at $\geq 3\times$ chance with low entropy	(a) in 0 of 9 at both lengths; prefix rate 0.08–0.15, chance 0.11 (L=8)	not supported
The transplant moves the answer position into a second stable state (multistability)	Multistability of the frozen-upstream map: five starts on the segment between sham and transplant, all probed lags, roster, cells where all five converged	multi-label cells 0/23,493 (L=8), 0/22,726 (L=6); readout and ℓ^* return by lag 24 in 285/286 and 274/274 erring pairs	not supported: one end label (end states not saved), moved and returned
The effective map warns before the readout errs	EM2, pre-declared; lead beyond the update built into the definition	EM2's second clause 2 of 9; asked two updates ahead, its first clause holds in 0 and 0 of 9; X1c beyond the built-in update in 1 and 0 of 6	not supported
The disturbance travels at one speed, whatever its size	Paired dose slopes (Pb′ re-analysis)	earlier with larger a in 7 of 9 (L=8), by 0.10–1.13 updates per doubling; at L=6 in 5 of 9, while $\mathbb{Z}_{61}$ s4 errs later by 1.12	rejected in this form
The disturbance is localised in time, shown by clamps	PK1's shifted-window clause	met in 9 of 9, but by construction (Appendix G.6); the timed clamp equals holding only at ceiling	not tested by PK1; tested later by R3a (Table 29)
The answer position's frozen map contracts in one step at its fixed point	X2a, pre-stated: largest singular value of its Jacobian below one in $\geq 95\%$ of roster cells	0.56 [0.50, 0.62] (estimates from below, so the share with the true value below one is at most this); medians above one in A_5 s0, D_{31} s0 and s1	not supported; convergence not recorded by this script; spectral radius not measured
Upstream evidence reaches the answer position at finite speed	Structure of the model	every operand token enters the answer position's attention from update 0; what is carried is the transplant's difference in a partly composed state, whose content EM5 does not identify	inconsistent with the architecture

Table 46: **Readings tested and not supported, second round** [X]. Rows 1–2, 4, 6 and 7 use predictions written before their runs; the other rows are post hoc, on data from Runs C, EM and G1c. Row 3 probes a counterfactual frozen-upstream map on one segment of starts. In row 4, "asked two updates ahead" is a post hoc variant. Row 6's clause holds almost by construction; a later script's finer shifts (R3a) do test localisation at M and find it in the robust four, so row 6 records a test that failed to inform, not a rejected reading. Row 7 is X2a (Table 28).

G.13 Open discrepancies between specification and implementation

We keep a ledger of places where a written specification (a plan, a script docstring or a draft sentence) and the code disagree. The items below concern this appendix. While any of them is open, we do not claim that the mechanism analyses implement their specifications exactly. Items about experiments not reported in this paper are omitted.

- **Readout position (open, measured).** The working hypothesis said that the readout errs when a disturbance reaches the answer position. The readout, however, applies the exit blocks to all positions and then reads the last one (Appendix G.1). Readout patches and cross readouts show that the route differs by checkpoint (Tables 35 and 38). The answer position's own state carries the wrong label in A_5 s0, D_{31} s0, s4 and $\mathbb{Z}_{61}$ s0, and the upstream positions are read directly in A_5 s3 (and in A_5 s2 and $(\mathbb{Z}_2)^6$ s0). A_5 s1, s4 and $\mathbb{Z}_{61}$ s4 are mixed, and D_{31} s1 is dispersed. We therefore write "readout-relevant positions", and ℓ^* mixes both routes.
- **Pair filter of the edit-position sweep (open).** The sweep slopes first reported, which also fixed the roster, were computed with a filter on the sham alone (sham correct at lag 0), not on both-correct pairs. The sham-only filter reproduces 22 of the 22 reported values and the both-correct filter 18. We use only both-correct values (Table 36).
- **Roster (open).** With the both-correct sweep, only 4 of the 9 roster checkpoints are chain-type under every candidate type rule with room to spare (rules and selection in Appendix G.3). We keep the declared roster for scoring and report the robust four alongside; the choice among the type rules is open.

- **Pa's exception clause (open).** The Run C docstring names A_5 s0, D_{31} s0 and $\mathbb{Z}_{61}$ s0 as exceptions to Pa (all-but-seam error $\geq 0.5\times$ full). The analysis script first scored only the main clause and reported Pa as held. The exception clause fails (D_{31} s0 is below 0.5 at both lengths), and $\mathbb{Z}_{61}$ s4, not named, also misses; we report Pa as held in part. The first analysis of PK3 likewise conjoined clauses that the docstring states separately; we report both counts (Appendix G.3).
- **Planned versus implemented interception (open).** Run B implements a subset of the planned arms. A displaced seam and a decohered donor were not run; a cut of intermediate positions on the full transplant is approximated by cuts on the seam-only transplant; and the planned clamp-release and time-shifted cuts were added later, in Run C and G1c. Run B uses all both-correct pairs of blocks 0–3 instead of the planned 48 erring and 48 non-erring pairs from blocks 0–5, which those blocks cannot supply. P1–P3 refer to the arms as implemented.
- **Direction-control pairs (open).** The plan asked the direction battery to use the same pairs as Run B, 48 erring and 48 non-erring pairs from blocks 0–5, which those blocks cannot supply. The battery instead stratifies by whether the archived donor errs and draws from all evaluation blocks, with fewer non-erring pairs in A_5 s3 (Appendix G.9); its pairs therefore differ from those of Runs A–C. The authors accepted the existing run after it had finished.
- **Direction-control hump ratio (open).** Two predictions of the direction battery use the ratio of the answer position's largest later displacement to its displacement at lag 0, which is exactly zero for every seam-only start, so their hump parts cannot be evaluated as written; no replacement has been adopted, and we report none of them as a verdict.
- **Power-iteration sample and steps (open, found after the run).** The delay-line script's docstring specifies the power iteration for the first 16 pairs per checkpoint with 15 steps. The script as run used 8 pairs at $L{=}8$ (none at $L{=}6$) and 10 steps, a change its own note dates after a smoke run and before the full run. The two edits of a context share the sham state, so the 8 pairs are 4 distinct states, each estimated twice; the two estimates differ by up to 0.24 of the larger. The docstring's preamble names both-correct pairs, but the power iteration ran on the first pairs without that filter; restricted to both-correct pairs, GC1's pooled clause falls below its threshold (0.89). We report the run, count each state once and treat every value as a lower bound (Appendix G.3).
- **Frozen-map sample (open, found after the run).** The preamble of the timed-clamp script's predictions names both-correct pairs; its frozen-answer-map statistics (X2a, X2b) were computed on the first 16 pairs of each checkpoint without that filter. Restricting them to the both-correct pairs leaves both verdicts unchanged (Appendix G.3); we report the unfiltered values as analysed.
- **Ties in R3a (recorded).** R3a asks for the largest removal at $d \in \{-1, 0\}$ without saying how a tie with $d = +1$ counts; the analysis took the first maximum. Counted strictly, R3a still holds at its declared length (Appendix G.3).
- **Shifted clamp windows (open).** PK1 treats the failure of windows shifted by ± 4 updates as evidence of localisation in time. Early windows mostly clamp a position the disturbance has not reached, and late windows mostly start at or after the first error, so that clause holds almost by construction (Appendix G.6). We do not claim localisation in time from it. The later windows shifted by two updates (R3a) are not forced this way, but they do not repair PK1's clause, which stays as written.
- **Nominal transmission (open).** PK3's docstring describes b as the share of the displacement transmitted at M. The code rescales the displacement after each update, and position M feeds back into itself, so the effective transmission is lower (Appendix G.6). The prediction text is unchanged; we report the effective values next to it.
- **Refinement along the donor segment (open).** The segment scan refines only intervals in which the labels of lags 1–8 change, at most the first 64 such intervals per level, and never intervals whose two ends agree. The single-switch share is therefore an upper bound over about 1.2 decades.
- **Fate-map specification (open).** The fate-map script departs from its plan in plane ranges, zoom levels, ε range and refinement depth, and it did not save the refined points of 134

refined intervals on long lines of $(\mathbb{Z}_2)^6$ s1 (multi-valued outcomes only; Appendix G.10). Its two-level falsification rule cannot be applied at the one admissible 16-fold level; the authors chose to apply it to a post hoc four-fold re-split of the saved points (Appendix G.10).

- **Fractal criteria that do not measure fractality (open).** The positive control shows that the scripts' threshold $\alpha < 0.85$ has no power against a boundary of dimension 1.2 at one 16-fold level (Appendix G.10). The ε-scan exponent does not measure an uncertainty exponent in this design (Appendix G.10). We use neither as evidence, and we rest the refinement result on unsplit coarse switches together with the positive control.
- **Propagator certificate (open).** The propagator's first-order certificate has opposite signs in the plan and in a Taylor expansion, and its fixed competitor often differs from the actual runner-up. We use only the exponents of that run.
- **Reuse experiment wording (open).** The E3 docstring and the first results note describe a new draw of k and contexts; the code draws only a new k, on the contexts of the sweep (Appendix G.7). The analysis script's settling statistic is the onset of the eight-readout run, not C_8 as defined in Section 2; we call it s_8.
- **Prefix reset in the reuse experiment (recorded; the authors approved amendment A1 after the outputs had been analysed).** The rationale of E5 called the prefix reset "scratch with an exact prefix"; the code joins the cached prefix (update t_{cache}) with initial values right of the edit (update 0). Amendment A1 corrected the rationale before any output was downloaded or opened (per the session log; the $L{=}8$ outputs already existed on the GPU host); the prediction text is unchanged.
- **Wording about arrival (preventive).** A description of the mechanism as upstream evidence arriving at finite speed conflicts with the architecture, in which every operand token is visible to the answer position from update 0. We describe the carried quantity as the transplant's difference in a partly composed state; what that state encodes is not identified (EM5).
- **A name (preventive).** A native settling time of the recipient was earlier given the same name as the margin-based quantity of the Lemma. The two quantities differ, and we keep their names apart.

H Training recipe and reproducibility details

New group-model replicas use batch size 256, AdamW with $(\beta_1, \beta_2) = (0.9, 0.95)$, learning rate 6×10^{-4}, 1,000 warm-up steps and cosine decay to one tenth of that rate. Weight decay is 0.1 for eligible matrix parameters and zero for vector/non-decayed parameters; gradient norm is clipped at 1. The first 40% of 30,000 steps expands the sampled sequence-length range from one to eight. Training uses full-vocabulary cross-entropy at the answer position, BF16 autocast and the configured `poisson-truncated-full` recurrent-depth sampler (mean recurrence 27, mean backpropagation depth eight). Thus training is not restricted to terminal argmax at a single fixed depth. Every 1,000 steps a checkpoint is evaluated in FP32 on 512 length-eight examples; strict improvement in update-27 full-vocabulary accuracy selects the saved checkpoint.

The architecture is Parcae (Prairie et al., 2026), built from a pinned commit of its `parcae_lm` code, with six attention and key/value heads, intermediate width 3,072, ReLU-squared MLPs, RMSNorm with epsilon 10^{-5}, query/key normalization, diagonal prompt injection and tied embeddings. It uses two blocks in each prelude, shared recurrent core and exit. Native initialization is generated with the configured `like-init` rule, not a zero vector. The evaluation scripts preserve the exact checkpoint configuration and initialization seed. New numerical and capability decisions retain all failed cells; the reference checkpoints' historical training-attempt denominator has not been reconstructed.

I Discovery-stage recovery and construction results

The following results motivated the independently registered follow-up in the main text. They are retained as a separate discovery archive. In the four-reference statistical experiment, opposite-fold

Table 47: **Computed configurations reduce recovery cost.** Capped horizon-stable donor delays and paired control-minus-donor contrasts. Intervals are the original simultaneous context-bootstrap bands with group correction. Recovery is measured in reference checkpoints only.

Group	Donor mean	Native minus donor	Random minus donor
$\mathbb{Z}_{61}$	10.227	-0.312 [-0.786, +0.161]	+1.898 [+1.465, +2.332]
$(\mathbb{Z}_2)^6$	7.477	+1.156 [+0.640, +1.673]	+1.156 [+0.640, +1.672]
D_{31}	12.453	+0.727 [+0.021, +1.432]	+1.832 [+1.285, +2.379]
A_5	10.621	+1.449 [+1.125, +1.773]	+2.688 [+2.340, +3.035]

fitted means and pooled scalar residual variance give

$$\widetilde{h}_i = \mu_{-f(i)} + \sigma_{-f(i)} z_i, \quad z_i \sim \mathcal{N}(0, I_n), \qquad h_i^{\text{mean+iso}} = \|d_i\|_2 \frac{\widetilde{h}_i}{\left\|\widetilde{h}_i\right\|_2}. \tag{5}$$

The old fitted pools recur across evaluated contexts; the shared-pool limitations below apply to these archived bands, not to the new independent-block sampling design.

I.1 Useful donors start with worse recipient readouts

In the reference experiment, donors shorten S_{27} relative to isotropic states by 1.90, 1.16, 1.83 and 2.69 updates for Z61, XOR, D31 and A_5; all original simultaneous intervals are positive (Table 47). Native initialization takes longer in three tasks. The Z61 native-minus-donor contrast is -0.31 with an interval crossing zero.

These benefits are not explained by initially correct donor outputs. Recipient predictions are wrong in 1,022/1,024 insertions. Their target margins average $-26.30, -26.99, -18.53, -20.66$, worse than both controls in every task. Excluding the two initially correct cases retains a positive donor-versus-random contrast in all four tasks. The unselected comparison remains primary; conditioning on a post-intervention margin would answer a different causal question. Appendix M gives complete lag-zero readouts and strata.

I.2 An advantage can emerge after the first correct answer

All eight native/random-minus-donor mean post-arrival contributions are positive (Figure 11). Against isotropic states they are 1.76, 1.59, 1.07 and 1.51 updates. XOR illustrates why the time definition matters: random states first reach the answer 0.430 updates earlier in the observed means, but their correct suffix starts 1.156 updates later. The difference is a 1.586-update larger post-arrival gap. The descriptive first-hit interval $[-0.906, 0.047]$ includes zero; the mean-ordering reversal is not a significant first-hit effect. In Z61, native initialization's 1.387-update arrival advantage offsets the donor's 1.074-update post-arrival advantage.

This pattern is not solely unequal observation opportunity after the first hit. For fixed post-hit windows of four and eight updates, retaining pairs where both hits occur early enough, all sixteen task/control/window mean wrong-label-count contrasts favor donors, by 0.300–1.242 updates. These post hoc selected comparisons are descriptive, not mediation estimates. Random-minus-donor S_H means also remain positive at every tested $H \in \{12, 16, 20, 24, 27\}$. At $H = 27$, four D31 records per condition remain nonreturns and are retained at the cap. Appendix M reports all windows and failures.

I.3 Discovery-stage activation statistics and mixed-state controls

The discovery archive asked which summary statistics of computed states suffice for a good continuation start. The next subsection compares mean-centered constructions with real donors in the four reference checkpoints; the two after it extend the mixed-state comparison to sixteen capable checkpoints and state which position, training and distance choices define the comparison.

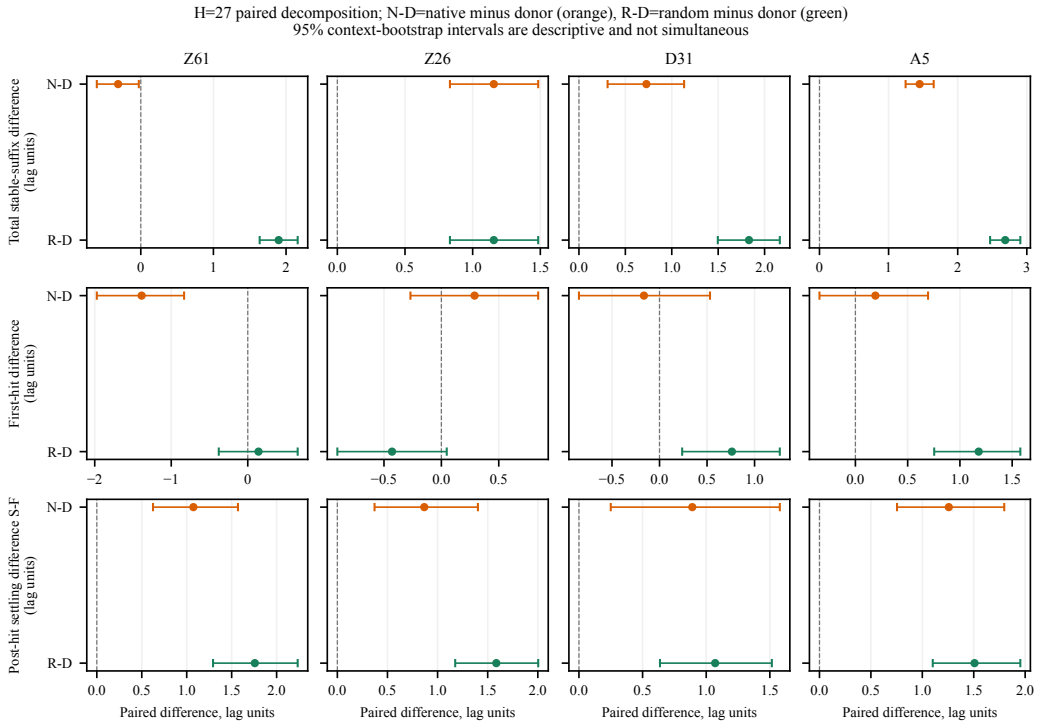


Figure 11: **First arrival and sustained correctness measure different intervention effects.** Each control-minus-donor sustained-delay difference equals its arrival and post-arrival components. Positive values favor donors. Error bars are descriptive 95% context-bootstrap intervals, distinct from the primary simultaneous bands in Table 47.

I.4 Mean-centered synthetic states outperform computed donors

The statistical-construction experiment holds the recipient continuation and donor norm fixed while changing the starting configuration. Mean plus isotropic residual in Eq. (5) reduces mean S_{27} relative to the real donor by 1.41, 0.69, 1.10 and 0.47 updates; all four archived resampling bands exclude zero (Figure 12). The ordering also holds separately in both held-out folds of every task. These are paired context results at one reference checkpoint and one calibrated intervention stage per task.

Positionwise norm profiles alone leave delays close to isotropic initialization. Introducing a mean-centered configuration changes the ordering: a state built without the paired donor's full computed configuration becomes a better continuation start. The result demonstrates the sufficiency of this *statistical construction* for outperforming the donor under the measured conditions. It does not establish that the mean alone is sufficient, that the scale carries no context information, or that the mean is optimal.

Covariance variants support a narrower conclusion than a monotone "more detail is worse" account. Mean plus diagonal covariance is faster in point estimate than mean plus isotropic noise in three tasks. Low-rank covariance constructions often move closer to the donor, but these comparisons change noise orientation and amplitude jointly. No adjacent-rung superiority test or general penalty for higher moments is inferred from the donor-relative bands. Appendix J reports the exact constructions, first-hit outcomes and all numerical contrasts.

Faster overall, with a larger post-arrival gap. Applying Eq. (2) to the synthetic comparison gives a complementary temporal result (Table 48). Mean plus isotropic states first reach the recipient answer 1.49–2.19 updates earlier than donors, but their capped post-arrival gaps are 0.77–1.03 updates larger. A post hoc, twelve-contrast resampling analysis retains these signs, subject to the

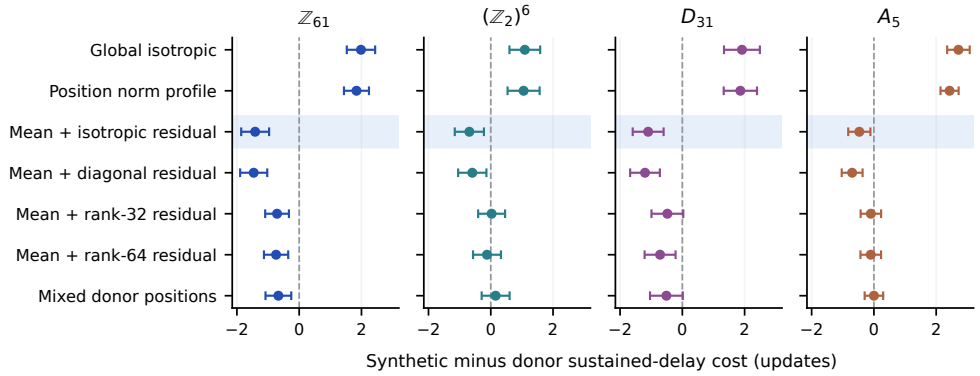


Figure 12: **Mean-centered statistical initializations can outperform real donors.** Control-minus-donor S_{27} contrasts in four reference checkpoints. The mean-centered condition includes isotropic residual noise and donor-specific norm rescaling. Bands summarize max-t saved-context resampling over seven contrasts with four-task correction; they do not establish coverage for the shared-pool construction. The categories are different constructions, not an ordered information hierarchy.

Table 48: **Mean-centered states arrive sooner but stabilize less promptly after arrival.** Mean-plus-isotropic minus donor, in updates. Post hoc max-t resampling bands over three outcomes per task with four-task correction; 50,000 draws. Shared-pool dependence is not reproduced, so procedure-level coverage is not established. The terms obey $\Delta S = \Delta A + \Delta B$.

Group	ΔA (first hit)	ΔS (stable)	$\Delta B = \Delta(S - A)$
Z61	-2.188 [-2.857, -1.518]	-1.414 [-1.838, -0.991]	+0.773 [+0.212, +1.335]
XOR	-1.488 [-2.104, -0.873]	-0.688 [-1.127, -0.248]	+0.801 [+0.278, +1.324]
D31	-1.941 [-2.755, -1.128]	-1.098 [-1.598, -0.597]	+0.844 [+0.078, +1.609]
A5	-1.500 [-1.960, -1.040]	-0.469 [-0.805, -0.132]	+1.031 [+0.524, +1.538]

shared-pool limitation above. Every task's two held-out folds also retain all three mean signs (Appendix J). Thus their faster sustained recovery is driven arithmetically by earlier arrival despite a larger subsequent gap. The better total-cost initialization is not the one with the smallest post-arrival gap. This decomposition concerns the recorded endpoint, not causal mediation; four D31 pairs per condition remain capped.

I.5 Mixed-state effects vary across sixteen capable checkpoints

Mixing trained vectors across donor-pool entries preserves their positional roles but removes the requirement that all positions come from one computation. Across sixteen checkpoints, the archived pointwise intervals indicate shorter S_{27} than the real donor in three, longer S_{27} in four, and overlap with zero in nine (Figure 13). The four positive mixed-minus-donor effects are XOR/s1 $+3.52$, XOR/s3 $+0.62$, Z61/s1 $+0.35$ and Z61/s3 $+0.54$ updates. Donors outperform native and isotropic states in twelve and thirteen checkpoints, respectively, under the same pointwise criterion.

The mixed-state contrast is heterogeneous, not a universal null. A positive effect can reflect useful cross-position dependencies or distributional compatibility; it does not isolate a transferred algorithmic intermediate. Conversely, an interval containing zero does not show that all donor computation is dispensable. These control results extend the recovery comparison across capable training realizations, while the synthetic-statistics result in Section I.4 remains a four-checkpoint experiment.

I.6 Position, training and distance specify the comparison

All-position permutation costs 0.38, 0.57, 1.06 and 0.36 updates in the reference models, with positive simultaneous bands in three tasks. Untrained-network states cost 2.18, 1.00, 2.92 and 2.51 updates, with positive bands in all four. Operand-only permutation shows no detected slowing. The

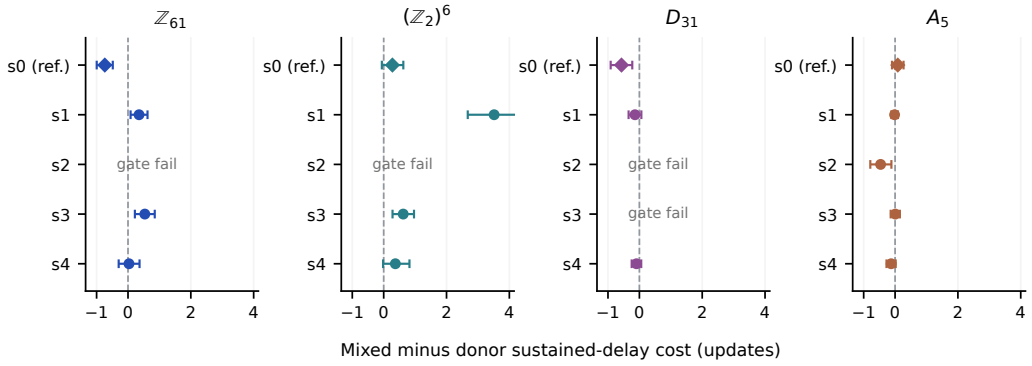


Figure 13: **The mixed-minus-donor contrast depends on the checkpoint.** Sustained-delay contrasts for sixteen capability-qualified models. Positive values favor the intact donor; negative values favor positionwise mixtures. Intervals are descriptive pointwise 95% saved-context bootstraps, uncorrected across checkpoints. The nine intervals crossing zero do not establish equivalence. Reference seeds are marked separately from additional training runs.

all-position intervention additionally changes both BOS and the answer slot, so this difference does not uniquely localize the effect to the answer position. The all-position controls were added by a documented amendment after the operand-only results; we retain both families (Appendix J).

Norm matching also leaves displacement from the recipient trajectory uncontrolled. Donors share six operands with recipients and can lie closer to that trajectory. In a separate 64-context-per-task diagnostic, scaling random displacements to the donor's displacement norm reverses the random–donor mean ordering in all four tasks. Because that construction uses the recipient's already computed state, it is a diagnostic rather than a deployable initialization. Equal norm therefore does not isolate an effect of donor computation, and the statistical interventions do not by themselves establish distance-independent benefits. Appendix K gives the paired construction and measured delays.

J State-construction controls and their audit

This appendix retains the preceding discovery-stage statistical-construction and mixed-state control results. Its saved-record analyses are separate from the independent-pool mean-plus-isotropic replication in the main text. The construction source hash matches its recorded run manifest. In particular, the condition named `pos_mean` adds an isotropic residual before global norm rescaling. Describing it as a deterministic mean-only state would change the experimental intervention.

Cross-fitting and synthetic conditions. For each reference checkpoint, pair indices 0–127 and 128–255 correspond to contexts 0–63 and 64–127, with both actions together. The opposite fold provides 128 state samples of shape 10×768. At position j, let μ_j be their mean, $s_{j,d}$ the sample coordinate standard deviation and r_j the mean position norm. The pooled residual variance used in Eq. (5) averages squared deviations over samples, positions and coordinates. The variants before final whole-state norm matching are:

- Global isotropic: a standard Gaussian state.
- Position norm profile: an isotropic vector at each position rescaled to r_j.
- Mean plus isotropic: $\mu_j + \sigma z_j$, using the pooled variance.
- Mean plus diagonal: $\mu_j + s_j \odot z_j$.
- Mean plus rank-k: $\mu_j + U_{j,k}\Lambda_{j,k}^{1/2}\xi_j + \sigma_{j,\mathrm{res}} z_j$, with $k = 32$ or 64 and residual variance equal to omitted covariance trace divided by width.

Every construction is subsequently scaled to the paired donor's full-state norm. Covariance variants use finite-sample estimates, independent positional draws and residual noise; they do not preserve the full cross-position covariance of a real computation. Their donor-relative effects do not identify a monotone hierarchy or statistical minimality. Table 49 reports all sustained-delay contrasts.

Shared pools and inferential target. Context bootstrapping moves both paired actions together. The archived synthetic-state bands use 50,000 studentized max-t draws over seven contrasts per task and a four-task Bonferroni correction (per-task tail 0.0125). The two fitted distributions, random draws and mixed-state pool are held fixed. Every evaluation context also contributes to the opposite fold's fitted construction. The saved-context bootstrap does not reproduce this dependence; holding pools fixed does not establish conditional independence or nominal simultaneous coverage. The intervals are resampling summaries, not coverage guarantees for the fitted-construction procedure. Neither conservatism nor anti-conservatism has been demonstrated. Population-level inference would require an analysis of the full construction and evaluation procedure.

As a descriptive sensitivity check, we reconstruct the paired contrasts separately for each held-out fold (64 contexts, two actions each). Every task-fold has lower mean arrival and sustained times for mean-plus-isotropic states, and a larger post-arrival gap. Table 50 rules out an opposite-sign-fold explanation of the pooled result; it does not establish population significance or interval coverage.

The synthetic-state records retain first-hit and suffix event flags as well as capped times, but not complete logit trajectories. The original reference recovery records retain all 28 logit vectors and admit a separate direct reconstruction. The sixteen-checkpoint extension stores arrays of capped first-hit and sustained times only. In that extension, an endpoint value of 27 cannot distinguish a first stable event exactly at the cap from no event; no event-rate conclusion is inferred from those arrays.

What mixing preserves. For each position, the implementation draws a derangement of the 256 action-pair indices. It excludes the exact receiving pair, but it can select the other action from the same context. Source contexts need not be distinct across positions. "Mixed" therefore means a positionwise composite of non-self donor records, not a guarantee of ten distinct problems. Whole-state norm rescaling additionally means that exact positionwise marginal distributions are not preserved after construction. The original control run, synthetic construction and sixteen-checkpoint extension use separately seeded pools, so their mixed-state effects need not match numerically.

Original and amended controls. The first registered controls permuted or mixed only the eight operand positions, leaving BOS and the answer position intact. An amendment added all-position versions while retaining the initial outcomes. The amended reference analysis uses max-t over five contrasts with four-task correction. The all-position shuffle is slower in three of four tasks, whereas the all-position mixture is faster in Z61 and D31 and has intervals crossing zero in XOR and A5. All four untrained-state contrasts are slower. This fails the original positive-coherence prediction; it does not establish that no computed dependencies matter. Nor does changing BOS and the answer slot together localize the effect uniquely to one slot. The untrained comparison uses one fixed random network initialization per task and is not a replication over untrained networks.

Sixteen-checkpoint extension. Table 52 retains every capable model. Checkpoint hashes agree with archived calibration identities. The runner's t_{80} roster matches those independently calibrated depths. The same 128 contexts per task recur across seeds, and the two actions are kept together. Reported original bands use 20,000 percentile context bootstraps separately for each of three contrasts in sixteen cells (48 pointwise comparisons). An independent 50,000-draw reconstruction reproduces all point estimates and the mixed-control tally, but does not add familywise validity. The extension is reported as exploratory; the four-reference protocol is not a preregistration of these 48 comparisons. No equivalence margin or equivalence test was specified.

K DISTANCE-MATCHED DIAGNOSTIC

An exploratory subset contains 64 contexts and two actions each per reference checkpoint. Let $h_\star$ be the recipient state already computed at t_{80}, d the donor and r the globally norm-matched random

Table 49: **All synthetic-state contrasts.** Capped S_{27} means and synthetic-minus-donor resampling bands. Shared-pool coverage is not established. The mean-centered condition includes isotropic noise and donor-specific scale.

Task	Construction	Mean	Contrast [band]
$\mathbb{Z}_{61}$	Global isotropic	12.211	+1.984 [+1.529, +2.440]
	Position norm profile	12.066	+1.840 [+1.438, +2.242]
	Mean + isotropic residual	8.812	-1.414 [-1.862, -0.966]
	Mean + diagonal residual	8.766	-1.461 [-1.896, -1.026]
	Mean + rank-32 residual	9.516	-0.711 [-1.092, -0.330]
	Mean + rank-64 residual	9.484	-0.742 [-1.132, -0.353]
	Mixed donor positions	9.559	-0.668 [-1.082, -0.254]
$(\mathbb{Z}_2)^6$	Global isotropic	8.570	+1.094 [+0.600, +1.587]
	Position norm profile	8.531	+1.055 [+0.540, +1.570]
	Mean + isotropic residual	6.789	-0.688 [-1.156, -0.219]
	Mean + diagonal residual	6.883	-0.594 [-1.048, -0.140]
	Mean + rank-32 residual	7.504	+0.027 [-0.405, +0.460]
	Mean + rank-64 residual	7.355	-0.121 [-0.569, +0.327]
	Mixed donor positions	7.633	+0.156 [-0.292, +0.605]
D_{31}	Global isotropic	14.367	+1.914 [+1.338, +2.490]
	Position norm profile	14.316	+1.863 [+1.329, +2.398]
	Mean + isotropic residual	11.355	-1.098 [-1.594, -0.602]
	Mean + diagonal residual	11.254	-1.199 [-1.676, -0.722]
	Mean + rank-32 residual	11.973	-0.480 [-0.992, +0.031]
	Mean + rank-64 residual	11.738	-0.715 [-1.212, -0.218]
	Mixed donor positions	11.941	-0.512 [-1.038, +0.015]
A_5	Global isotropic	13.336	+2.715 [+2.353, +3.077]
	Position norm profile	13.051	+2.430 [+2.139, +2.721]
	Mean + isotropic residual	10.152	-0.469 [-0.825, -0.112]
	Mean + diagonal residual	9.922	-0.699 [-1.034, -0.364]
	Mean + rank-32 residual	10.523	-0.098 [-0.427, +0.231]
	Mean + rank-64 residual	10.520	-0.102 [-0.432, +0.229]
	Mixed donor positions	10.621	+0.000 [-0.297, +0.297]

Table 50: **Both held-out folds retain the pooled mean ordering.** Mean-plus-isotropic minus donor, in updates. Each fold contains 64 contexts and 128 paired-action records. Point estimates only; no significance or coverage claim is inferred from this check.

Task	Fold 0			Fold 1		
	ΔA	ΔS	ΔB	ΔA	ΔS	ΔB
Z61	-1.9922	-1.5625	+0.4297	-2.3828	-1.2656	+1.1172
XOR	-1.5938	-0.7031	+0.8906	-1.3828	-0.6719	+0.7109
D31	-2.2734	-1.1719	+1.1016	-1.6094	-1.0234	+0.5859
A5	-1.5312	-0.4844	+1.0469	-1.4688	-0.4531	+1.0156

state. The diagnostic constructs

$$r' = h_\star + \frac{\|d - h_\star\|}{\|r - h_\star\|}(r - h_\star), \qquad d' = h_\star + \frac{\|r - h_\star\|}{\|d - h_\star\|}(d - h_\star).$$

Thus r' matches donor displacement and d' matches random displacement. Both use the already computed recipient state and cannot be treated as computation-free deployment baselines. The raw subset contains saved capped delays and displacement norms; it does not store the complete output traces or event flags. The table reports means without inferential intervals.

The reversal shows that donor superiority over globally norm-matched isotropic states is not invariant to the comparator's distance from the recipient trajectory. It does not show that distance alone explains all effects: at the donor's radius, the random direction can outperform the donor substantially. The statistical-state conditions also change distance and orientation together. Their positive result is the measured performance of specified constructions, not a distance-adjusted effect of one abstract state attribute.

L Observation maps and what output behavior identifies

The finite-horizon centered observation map and its derivative kernel are

$$\mathcal{O}_{c,H}(h) = \big(P\ell_c(h), \dots, P\ell_c(F_c^H h)\big), \quad P = I - \mathbf{1}\mathbf{1}^\top/V. \tag{6}$$

With $\mathcal{N}_{c,H}(h) = \ker D\mathcal{O}_{c,H}(h)$, the chain rule gives the standard observability relation (Hermann & Krener, 1977)

$$\mathcal{N}_{c,H}(h) = \bigcap_{t=0}^{H} \ker\big[PD\ell_c(F_c^t h)DF_c^t(h)\big] \subseteq \ker[PD\ell_c(h)]. \tag{7}$$

Proposition 3 (First-order visibility under continuation)**.** *For C^1 maps $F_c : \mathbb{R}^n \to \mathbb{R}^n$ and $\ell_c : \mathbb{R}^n \to \mathbb{R}^V$, the map in Eq. (6) satisfies Eq. (7) and*

$$\dim \mathcal{N}_{c,H}(h) \geq \max\{0, n - (H+1)(V-1)\}.$$

If $v \in \mathcal{N}_{c,0}(h) \setminus \mathcal{N}_{c,H}(h)$, its perturbation has zero first-order effect on current centered logits and a nonzero first-order effect on at least one future centered-logit vector.

Proof. The derivative of the stacked observation map stacks $PD\ell_c(F_c^t h)DF_c^t(h)$ over $t = 0, \dots, H$. Its kernel is their intersection. Each block has rank at most $V - 1$, giving the lower bound by rank–nullity. The final claim follows from the first-order expansion in direction v. □

The dimension bound alone does not establish that the set difference is nonempty. Let $Q(h) = P\ell_c(h)$ and let $R_H(h)$ stack the future blocks. Linear algebra gives

$$\operatorname{rank} D\mathcal{O}_{c,H}(h) - \operatorname{rank} DQ(h) = \operatorname{rank}\big(DR_H(h)|_{\ker DQ(h)}\big)\,.$$

Positivity is equivalent to present-invisible, future-visible first-order sensitivity. If this increment is positive and Q has constant rank near h, the constant-rank theorem gives a curve $\gamma(\epsilon)$ through

Table 51: **Reference controls and their retained amendment.** Control-minus-donor S_{27}. Bands: five contrasts per task, four-task correction. Positive values favor the donor.

Task	Construction	Contrast [band]
$\mathbb{Z}_{61}$	Operand permutation	-0.270 [-0.643, +0.104]
	Operand mixing	-0.902 [-1.332, -0.473]
	All-position permutation	+0.383 [-0.031, +0.796]
	All-position mixing	-0.770 [-1.187, -0.352]
	Untrained-network state	+2.176 [+1.743, +2.609]
$(\mathbb{Z}_2)^6$	Operand permutation	+0.246 [-0.253, +0.745]
	Operand mixing	+0.207 [-0.260, +0.674]
	All-position permutation	+0.566 [+0.101, +1.032]
	All-position mixing	+0.148 [-0.312, +0.609]
	Untrained-network state	+0.996 [+0.503, +1.489]
D_{31}	Operand permutation	+0.465 [-0.082, +1.012]
	Operand mixing	-0.426 [-0.977, +0.125]
	All-position permutation	+1.059 [+0.477, +1.640]
	All-position mixing	-0.766 [-1.266, -0.266]
	Untrained-network state	+2.918 [+2.346, +3.490]
A_5	Operand permutation	+0.281 [-0.087, +0.650]
	Operand mixing	+0.035 [-0.263, +0.333]
	All-position permutation	+0.355 [+0.032, +0.679]
	All-position mixing	-0.043 [-0.308, +0.223]
	Untrained-network state	+2.512 [+2.133, +2.890]

Table 52: **Sixteen capability-qualified checkpoint controls.** Control-minus-donor S_{27} differences. Native/random columns are point estimates; mixed resampling intervals are pointwise 95%, uncorrected across the 48 examined contrasts. Shared-pool coverage is not established. Seed 0 denotes a reference model.

Task / seed	t_{80}	Donor	Native	Isotropic	Mixed [interval]
$\mathbb{Z}_{61}$ / s0	11	10.227	-0.312	+1.844	-0.738 [-0.996, -0.484]
$\mathbb{Z}_{61}$ / s1	11	7.621	+2.105	+3.520	+0.352 [+0.082, +0.621]
$\mathbb{Z}_{61}$ / s3	6	4.156	+1.383	+3.922	+0.535 [+0.219, +0.852]
$\mathbb{Z}_{61}$ / s4	10	10.164	-1.266	-0.641	+0.031 [-0.297, +0.367]
$(\mathbb{Z}_2)^6$ / s0	10	7.477	+1.156	+1.156	+0.277 [-0.059, +0.625]
$(\mathbb{Z}_2)^6$ / s1	10	8.617	+0.281	-1.527	+3.516 [+2.676, +4.395]
$(\mathbb{Z}_2)^6$ / s3	10	7.297	+1.055	+2.785	+0.621 [+0.281, +0.965]
$(\mathbb{Z}_2)^6$ / s4	15	9.672	+3.094	+3.355	+0.371 [-0.020, +0.820]
D_{31} / s0	15	12.453	+0.727	+1.645	-0.570 [-0.914, -0.226]
D_{31} / s1	12	11.180	-0.031	-0.312	-0.141 [-0.344, +0.062]
D_{31} / s4	12	9.910	+0.957	+2.141	-0.094 [-0.246, +0.059]
A_5 / s0	13	10.621	+1.449	+2.523	+0.086 [-0.098, +0.270]
A_5 / s1	12	9.023	+2.016	+2.062	-0.016 [-0.121, +0.090]
A_5 / s2	13	11.754	+0.645	+2.820	-0.461 [-0.789, -0.117]
A_5 / s3	12	9.812	+1.844	+1.430	+0.008 [-0.141, +0.152]
A_5 / s4	13	8.801	+2.980	+3.207	-0.125 [-0.273, +0.020]

Table 53: **Displacement matching changes the random–donor comparison.** Exploratory 128-pair subset per reference task, $H = 27$. This diagnostic is separate from the cross-fitted statistical-construction experiment.

Start	Z61	XOR	D31	A_5
Donor d	10.273	7.344	12.656	10.555
Global-norm random r	12.266	8.336	14.172	13.242
Random displacement, donor radius r'	0.727	1.742	8.633	10.328
Donor displacement, random radius d'	12.477	8.867	13.758	12.188

h, tangent to such a direction, with $Q(\gamma(\epsilon)) = Q(h)$ exactly and $\frac{d}{d\epsilon} R_H(\gamma(\epsilon))|_0 \neq 0$. Thus the distinction can extend beyond first order under the stated local assumption. We have not measured this rank increment in the trained models.

For linear state and readout maps, the relevant kernel is $\bigcap_{t=0}^{H} \ker(CA^t)$, not just $\ker C$. For example, $C = (1\ 0)$ and $A = \left(\begin{smallmatrix} 0 & 1 \\ 0 & 0 \end{smallmatrix}\right)$ make the second coordinate invisible now and directly visible next update. Conversely, first-order invisibility at one state says nothing about arbitrary finite displacements, other prompts or training gradients in shared parameters. The large state dimension therefore does not imply unconstrained training dynamics. The objective uses cross-entropy at sampled depths and weight decay, not a single terminal argmax.

An output-equivalent construction with different gains. Write $h = (z, u)$ and consider

$$F_{c,s}(z, u) = (f_c(z), g_{c,s}(z, u)), \qquad \ell_c(z, u) = r_c(z).$$

For a common z_0, every differentiable choice of $g_{c,s}$ gives the same complete output sequence, while

$$DF_{c,s} = \begin{pmatrix} Df_c & 0 \\ D_z g_{c,s} & D_u g_{c,s} \end{pmatrix}$$

can change. Replacing $u \mapsto \frac{1}{2}u$ with $u \mapsto \left(\begin{smallmatrix} 1/2 & 2 \\ 0 & 1/2 \end{smallmatrix}\right) u$ introduces Euclidean transient growth without changing outputs. This is a non-identification construction, not a claim that our architecture has that block structure or that seed variability must occur.

Readout dimensionality also imposes no ordering of starting states. The scalar contraction $x \mapsto (x + 1)/2$, with correct label when $x \geq 0.9$, reaches that label in three updates from $x = 0.5$ and four from $x = 0$. Arbitrarily many unobserved coordinates can be added without changing either time. Thus the measured initialization advantage is empirical, not a consequence of a null-space count.

M JOINT READOUT–RECOVERY RECONSTRUCTION

The discovery-stage analysis reads all 1,024 saved pair JSON files from the second overnight recovery run. Each file contains 28 full logit vectors per condition (insertion plus 27 updates). An independent direct-array reconstruction checks every predicted label against the full-vocabulary argmax and reconstructs capped first-hit and suffix means. Context-level resampling retains the paired actions. The current readout classification is evaluated under recipient exit processing, rather than inferred from the donor prompt's target.

Wrong-only and donor-label-only strata are post-intervention descriptions. They do not estimate a controlled direct effect of a margin, and they do not replace the unselected primary population. Where only one of a context's two actions belongs to a stratum, the statistic is a ratio of selected-pair totals; bootstrap sampling draws whole contexts before recomputing that ratio. Complete paired values, failure flags, descriptive intervals and the five fixed horizon summaries accompany the analysis.

The copied primary confidence bands in Table 47 remain the original simultaneous bands. New descriptive pointwise intervals do not supersede their multiplicity control. The post-arrival quantity B_H includes a capped contribution for nonreturns; it is not a completed settling delay in those cases. Post-hit departures and subsequent returns are counted separately.

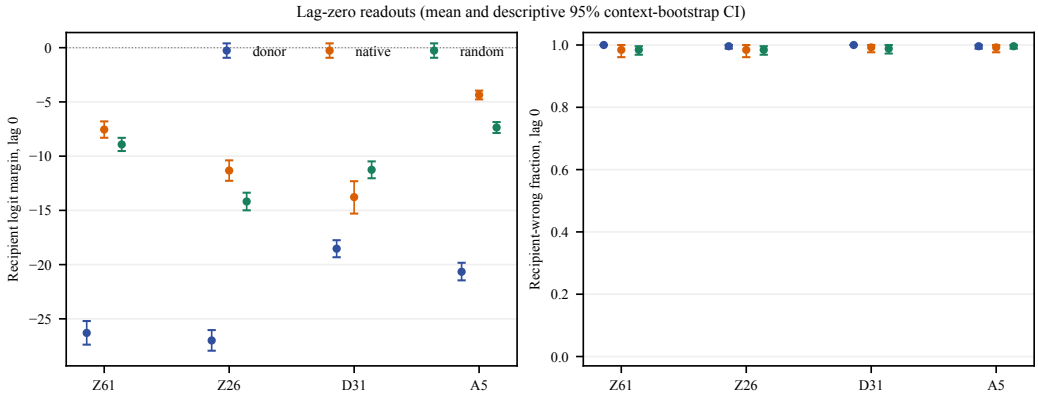


Figure 14: **Insertion-time readout is not continuation utility.** Target margins and wrong-answer fractions in the original 1,024 reference-checkpoint pairs. Intervals are descriptive context bootstraps. These logit traces are distinct from the synthetic-state endpoint records.

Table 54: Initially recipient-wrong donor strata. Contrasts are paired control-minus-donor means. Brackets give descriptive, nonsimultaneous 95% context-cluster ratio-bootstrap intervals. The primary unselected simultaneous intervals remain in Table 47.

Group	Wrong	Donor-label	Native S_H contrast	Random S_H contrast
Z61	256	216	-0.312 [-0.602, -0.023]	+1.898 [+1.629, +2.156]
Z26	255	213	+1.149 [+0.819, +1.480]	+1.149 [+0.820, +1.482]
D31	256	233	+0.727 [+0.297, +1.145]	+1.832 [+1.500, +2.160]
A5	255	218	+1.443 [+1.240, +1.647]	+2.690 [+2.471, +2.914]

N FINITE-BUDGET READOUT GEOMETRY

This section defines the Countdown state-plane observables of Figure 6: next-token partitions at fixed budgets, their finite-resolution boundary exponents, and the last-switch field. Figure 15 shows all five budgets; Appendices S and O give controls and counterexamples to a contraction reading. The intermediate partitions have uncertainty-derived effective exponents 1.430–1.442 over separations of 1–8 pixels and mixed-box exponents 1.392–1.446; smooth checkerboards can also yield high box slopes, and uncertainty slopes change over larger fit ranges, so these are finite-budget descriptions, not fractal dimensions.

N.1 INTRICATE INTERMEDIATE PARTITIONS, UNIFORM LATE READOUT

Figure 15 has 10, 3, 2, 2 and 1 distinct tokens at the five budgets. At 70, 140 and 280, the uncertainty estimates are 1.442, 1.442 and 1.430, while mixed-box estimates are 1.446, 1.433 and 1.392. Their apparent agreement is not independent evidence of fractality: both operate on the same finite image, and mixed-box fits also return large exponents for smooth periodic checkerboards. The uncertainty fit spans only $\log_{10} 8 = 0.90$ decades; its slope changes when the upper scale changes. We report effective exponents rather than a fractal limit.

The last-switch field is a different observable. It still has 332 distinct values at budget 560, even though the next-token partition is uniform. An instantaneous absence of a label boundary does not erase the history of when points changed labels. Nor does a small fraction switching exactly at an intermediate checkpoint prove that all other points will remain unchanged at later budgets.

N.2 SENSITIVITY AND READOUT MARGIN

An isotropic affine relaxation $F_c(h) = ah + (1-a)h_c^\star$ has $L_u(t,k) = k \log |a|$ for every direction, and hence $\Delta = 0$. Reliable nonzero contrasts exclude that scalar local description. An anisotropic

Table 55: Departures from the correct answer after a first hit, divided by the number of pairs with a first hit. A departure is counted once per pair; recurrent return-crossing counts are recorded separately in the released JSON. These are descriptive proportions.

Group	Donor	Native	Random
Z61	11/256	64/256	78/256
Z26	25/256	62/256	107/256
D31	64/255	82/254	79/256
A5	17/256	64/256	73/256

Table 56: Random-minus-donor mean capped suffix delay at five fixed horizons. Each value retains all 256 action pairs per group. Positive values favor donor initialization.

Group	$H = 12$	16	20	24	27
Z61	1.332	1.898	1.898	1.898	1.898
Z26	1.125	1.156	1.156	1.156	1.156
D31	0.629	1.609	1.824	1.832	1.832
A5	1.312	2.645	2.688	2.688	2.688

linear recurrence already escapes this constraint. Non-normal operators can produce transient norm growth despite stable eigenvalues (Hennequin et al., 2012); observing growth alone does not identify non-normality.

The link to a token map requires its readout. Let $m_t(h) > 0$ be the top-two logit margin at $F_c^t(h)$. The following bound separates state sensitivity from the output decision.

Proposition 4 (Sensitivity–margin stability)**.** *Suppose F_c^t is K_t-Lipschitz on a region containing h and $h + v$, in the Euclidean state norm, and ℓ_c is B-Lipschitz from that state norm to the maximum logit norm on their image. Then*

$$2BK_t \|v\|_2 < m_t(h) \quad \Longrightarrow \quad q_c(F_c^t(h + v)) = q_c(F_c^t(h)). \tag{8}$$

If L_j bounds each update on the respective propagated region, $K_t \le \prod_{j=0}^{t-1} L_j$.

Proof. Each logit changes by at most $BK_t \|v\|_2$. The winning logit can decrease and a competing logit increase by that amount, reducing their gap by at most twice the bound. A positive remaining gap preserves the winner. The product bound follows by composition. □

There is no converse from a uniform label to $K_t < 1$: unobserved coordinates can expand while all outputs lie inside one decision region. Conversely, even a strictly contracting smooth map can produce arbitrarily fine finite-budget label contours. For example,

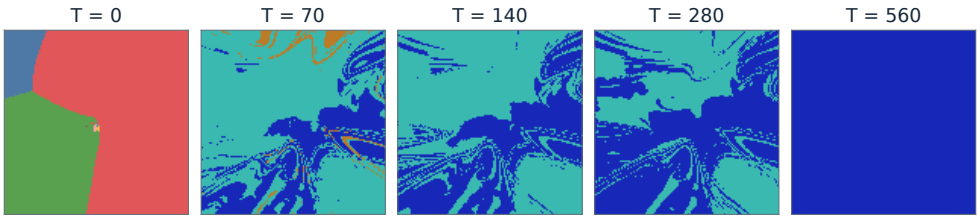
$$F_N(x, y) = \left(\frac{x}{4}, \frac{y}{4} + \frac{\sin(Nx)}{4N} \right) \tag{9}$$

has Euclidean Lipschitz constant below $1/2$, but the one-step preimage of the readout boundary $y = 0$ is $y = -\sin(Nx)/N$. Its wavelength can be arbitrarily small. Appendix O also gives a contracting Lipschitz map with an exactly fractal one-step partition and a constant two-step state. Thus neither the proposed necessity of expansion nor the proposed necessity of late global contraction follows from the images.

The resulting synthesis has three components. A useful starting configuration lowers recovery cost under a continuing input; recurrence changes selected state directions at unequal rates; and an output partition depends on those changes together with readout margins and geometry. A5's transient gain constrains its measured group-model trajectories. Countdown's uniform late plane constrains a different checkpoint's readout on sampled states. They cannot be identified as the expansion and contraction phases of a single measured map.

Table 57: Equal post-hit observation windows. Both conditions must first hit by $27 - w$. Entries are control-minus-donor paired differences with descriptive, pointwise 95% context-cluster ratio-bootstrap intervals. Counts report retained pairs, out of 256. Conditioning on both first hits makes this a descriptive robustness check.

Group	Control	w	Retained	Any departure	Wrong-label count
Z61	native	4	256	0.211 [0.133, 0.293]	0.566 [0.344, 0.801]
Z61	random	4	256	0.266 [0.203, 0.328]	0.777 [0.598, 0.969]
Z61	native	8	256	0.207 [0.129, 0.285]	0.766 [0.441, 1.113]
Z61	random	8	256	0.262 [0.203, 0.324]	1.242 [0.918, 1.574]
Z26	native	4	256	0.152 [0.074, 0.234]	0.480 [0.246, 0.727]
Z26	random	4	256	0.328 [0.254, 0.398]	0.934 [0.738, 1.141]
Z26	native	8	256	0.145 [0.066, 0.227]	0.688 [0.336, 1.070]
Z26	random	8	256	0.320 [0.246, 0.391]	1.195 [0.906, 1.481]
D31	native	4	253	0.075 [-0.020, 0.168]	0.300 [0.071, 0.529]
D31	random	4	255	0.055 [-0.020, 0.129]	0.357 [0.188, 0.533]
D31	native	8	253	0.075 [-0.016, 0.169]	0.585 [0.218, 0.992]
D31	random	8	254	0.059 [-0.016, 0.134]	0.709 [0.442, 0.984]
A5	native	4	256	0.184 [0.102, 0.266]	0.547 [0.336, 0.785]
A5	random	4	256	0.219 [0.152, 0.285]	0.648 [0.480, 0.824]
A5	native	8	256	0.184 [0.105, 0.270]	0.898 [0.527, 1.301]
A5	random	8	256	0.219 [0.152, 0.285]	1.000 [0.727, 1.289]



Measure	T=0	T=70	T=140	T=280	T=560
Distinct token labels	10	3	2	2	1
d_unc = 2 − α (OLS SE)	1.067 ± 0.004	1.442 ± 0.022	1.442 ± 0.025	1.430 ± 0.022	undefined
d_box (OLS SE)	0.921 ± 0.029	1.446 ± 0.048	1.433 ± 0.046	1.392 ± 0.046	undefined

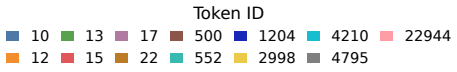


Figure 15: **A token partition can simplify while its earlier dynamics remain complex.** Direct model evaluations of the same 513×513 Countdown P4 plane at five budgets; colors consistently denote token IDs across panels. Effective uncertainty and box-count slopes are displayed with OLS residual SEs, over the stated pixel ranges. The intermediate uncertainty exponents correspond to $d_{\mathrm{unc}}^{\mathrm{eff}} = 1.430$–$1.442$. At $T = 560$, all sampled points emit token ID 1204, so boundary estimates are undefined. This is a finite-grid next-token observation, not proof of a unique full answer, a globally contracting state map, or the disappearance of settling-time structure.

O Counterexamples to a single Lipschitz explanation

Fine contours under smooth contraction. For Eq. (9), $N \geq 1$ and $(x, y) \in [-1, 1]^2$, the image remains in the square. Its Jacobian is

$$DF_N = \frac{1}{4} \begin{pmatrix} 1 & 0 \\ \cos(Nx) & 1 \end{pmatrix}.$$

The Euclidean operator norm is bounded by $(1+\sqrt{5})/8 < 1/2$. Nevertheless, thresholding the second output coordinate at zero gives the preimage boundary $y = -\sin(Nx)/N$, with arbitrarily short wavelength. This example has a smooth boundary of dimension one. It disproves an expansion requirement for fine finite-scale structure; it does not use that structure as evidence of a fractal limit. The amplitude and wavelength change together, illustrating why a bound on displacement alone cannot lower-bound separation between arbitrary readout contours.

An exactly fractal finite-budget partition under a Lipschitz contraction. Let $C \subset [0,1]$ be the middle-third Cantor set. On every removed interval of generation $n \geq 1$, set

$$g(x) = (-1)^n \operatorname{dist}(x, C), \qquad g(x) = 0 \quad (x \in C).$$

This g is 1-Lipschitz. On regions with the same sign the claim follows from the distance function's Lipschitz property. For opposite signs, a point of C between the arguments bounds the sum of their distances to C by their separation. Define

$$F(x,y) = (1/2, ag(x)), \qquad (x,y) \in [0,1] \times [-1,1], \quad 0 < a < 1.$$

Then $\mathrm{Lip}_2(F) \leq a < 1$ and F preserves the domain. With binary readout $q(x,y) = \mathbf{1}\{y > 0\}$, positive and negative gaps accumulate at every point of C. The one-step decision boundary is $C \times [-1,1]$, with box dimension $1 + \log 2/\log 3$. Since $g(1/2) = -1/6$, the two-step state is the constant $(1/2, -a/6)$. Thus an exactly fractal finite-budget readout partition followed by a uniform partition is compatible with strict contraction throughout. This counterexample is Lipschitz but not smooth; its purpose is to refute the claimed universal Lipschitz implication. The preceding smooth example covers arbitrarily fine finite-resolution contours.

A monochrome readout with expanding unobserved dynamics. On the invariant rectangle $[-1,1] \times [0,1]$, let

$$F(x,y) = (x/2, 4y(1-y)), \qquad q(x,y) = \mathbf{1}\{x > c\}, \quad 0 < c < 1.$$

For every t such that $2^{-t} < c$, all initial states have $q(F^t(x,y)) = 0$, with positive readout margin if logits are $(0, x-c)$. But at $y = 0$ the derivative of the second component of F^t is 4^t, so $\mathrm{Lip}_2(F^t) \geq 4^t$. This is a bounded, smooth, discrete-time system. Output uniformity does not require cumulative global contraction, even with a positive output margin.

What positive gain does imply. For a measured central-difference response L_u, the state-map Lipschitz constant on the sampled pair is at least $\exp L_u$. A positive *individual* gain therefore supplies a lower bound exceeding one. A positive mean log gain guarantees that some measured gains are positive, not that every point expands. Neither statement identifies non-normality: even a diagonal map can expand a direction. A stable non-normal matrix is one sufficient construction for short-window growth, but the experiments do not measure the operator or its commutator with its adjoint. Norm choice also matters for finite-time gains.

P Recovery definitions, controls and censoring

For labels $q_0, \ldots, q_H$ and target y, first-hit time is

$$R_{\mathrm{hit}} = \inf\{r \in \{0, \ldots, H\} : q_r = y\}.$$

No hit by H is administratively censored after reading q_H. An event exactly at H is counted as an event. Horizon-stable suffix onset is

$$R_{\mathrm{suffix,H}} = \inf\{r \in \{0, \ldots, H\} : q_s = y \text{ for all } s = r, \ldots, H\}.$$

It uses future labels to define an earlier onset, unlike a stopping time. When the set is empty we record nonreturn and use H in the capped mean. Consequently, $\mathbb{E}[\min(R_{\mathrm{hit}}, H)]$ is a conventional restricted mean, whereas the suffix mean is a similarly capped but explicitly retrospective summary. Both distinguish endpoint failure from a return occurring exactly at the cap.

The original equivalence prediction asks whether the simultaneous native-minus-donor interval lies inside $[-1,1]$ updates. It succeeds only for Z61. A positive difference and failure of equivalence are separate decisions. The random-minus-donor contrast is positive in all four checkpoints. Random minus native has mean zero in XOR, so the complete three-way rank order requested by a universal reuse interpretation does not hold.

Table 58: Six-condition recovery: terminal target agreement / capped suffix delay. Conditions c/d are timed from the switch (23-update horizon); the others from insertion (27 updates). All 256 pair records per group are included.

Group	a	b	c	d	e	f
$\mathbb{Z}_{61}$	1.000 / 10.23	0.996 / 0.31	0.996 / 9.99	1.000 / 10.24	1.000 / 9.91	1.000 / 12.12
$(\mathbb{Z}_2)^6$	1.000 / 7.48	1.000 / 0.25	1.000 / 7.40	1.000 / 7.49	1.000 / 8.63	1.000 / 8.63
D_{31}	0.984 / 12.45	0.977 / 0.84	0.977 / 12.11	0.984 / 12.38	0.984 / 13.18	0.984 / 14.29
A_5	1.000 / 10.62	1.000 / 0.26	1.000 / 10.31	1.000 / 10.64	1.000 / 12.07	1.000 / 13.31

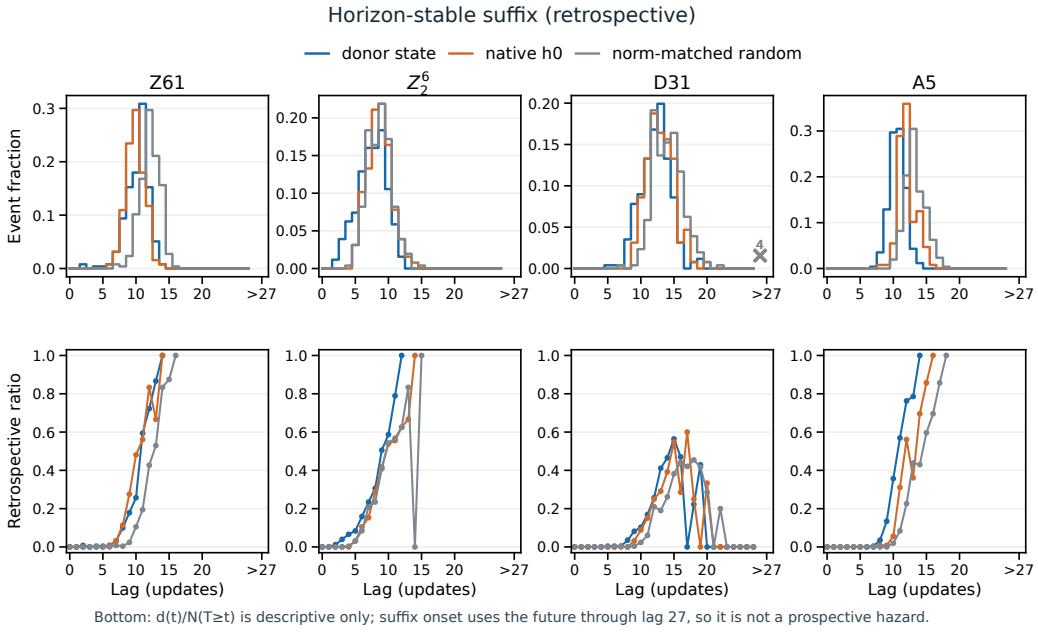


Figure 16: **Retrospective sustained-recovery summaries.** Same reference-checkpoint records as Table 47; the event is the onset of a target-label suffix lasting to lag 27. The curves are descriptive suffix-onset ratios, not prospective escape hazards. Nonreturns remain separately indicated. They reproduce the A5 lag-13 value $11/14$, whose denominator is retained in the machine-readable output.

Specified memoryless descriptions. A discrete geometric delay on positive integers has constant hazard p, mean $1/p$ and coefficient of variation $\sqrt{1-p}$, not exactly one. A continuous exponential has CV one. Continuous Weibull shape parameters obtained by applying a density likelihood to integer suffix times mix these models and ignore the retrospective event definition; we do not use those likelihood-ratio significance levels. Independently saved first-hit diagnostics instead use action 0 from each of 128 independent contexts, fixed thresholds 6/8/10, geometric tails with censoring, and a CDF discrepancy calibrated by 10,000 parametric bootstrap refits. Of the twelve corrected checks, only XOR at threshold 10 is not rejected. This does not reject a delayed escape process, an initialization-dependent hazard, or a mechanism with a different observed event.

The earlier elapsed-time comparison has a different population. The B2 clock analysis uses 174 common donor-switchable pairs and two later intervention stages, producing 348 comparisons. Truncating each earlier trace to the later remaining budget reproduces 313 returns and 35 nonreturns. These are identical terminal recovery indicators, not a verified equality of latent trajectories. In the newer six-condition experiment, pairs are unselected and receive 27 updates after transplantation. Mixing the two populations or budgets would create apparent contradictions in their mean delays.

Why a three-update discrepancy is not an identified mechanism. A scalar fitted contraction rate plus an arbitrary readout threshold does not uniquely predict answer-recovery time. Two different stable rates in an anisotropic linear system already generate distinct norm and readout time scales. A delay exceeding a scalar prediction therefore rejects that fitted description only after the prediction and threshold are specified; it does not measure three updates of active reconstruction or uniquely identify transport.

Q FULL SEED RESULTS AND NUMERICAL RECONSTRUCTION

The companion A2 experiment holds recipient input fixed after defining equal-norm task and answer-preserving directions from matched input actions. For $\delta = \epsilon \|h_t\|_2$, its finite-radius response is

$$L_u(t,k;\epsilon) = \log \frac{\left\|F_c^k(h_t+\delta u) - F_c^k(h_t-\delta u)\right\|_2}{2\delta}, \qquad \Delta = L_{\text{task}} - L_{\text{null}}. \tag{10}$$

The primary start is each seed's independently calibrated steepest accuracy increase; the primary window is one update. A numerical gate compares JVP and central-difference task-minus-null estimators. Here "null" is an answer-preserving input transformation, not a readout null direction. Among twelve new capable checkpoints, eight primary simultaneous bands exclude zero in the predicted direction, one excludes it in the opposite direction and three include it. XOR/s4 has $\Delta = 0.07895$ with interval $[0.03520, 0.12270]$. All four new A5 task gains are positive over one update, whereas the reference contracts. All measured four-update task means and simultaneous upper bounds are negative. Each new training seed calibrates its own milestones and peak using data separate from the measurement contexts. The pooled summaries contain sixteen measured checkpoints, including the four references; confirmatory new-seed summaries contain twelve. Their group counts are 3, 3, 2 and 4. These small numbers do not support a distribution-free generalization from all observed signs to every training realization. Four positive independent signs alone yield a one-sided sign-test probability $1/16$.

Table 59: Complete A2 primary results. Bands are simultaneous within each measured family with the recorded checkpoint correction; they are not seed-population confidence intervals.

Group	Seed	Start	n	L_{task}	L_{null}	Δ [simultaneous band]
$\mathbb{Z}_{61}$	0	9	128	-0.0505	-0.2637	+0.2132 [+0.180, +0.247]
$\mathbb{Z}_{61}$	1	7	128	-0.3631	-0.4383	+0.0752 [+0.035, +0.115]
$\mathbb{Z}_{61}$	2	–	–	–	–	Capability failure
$\mathbb{Z}_{61}$	3	4	128	-0.2431	-0.2877	+0.0446 [+0.001, +0.089]
$\mathbb{Z}_{61}$	4	7	128	-0.1662	-0.1539	-0.0123 [-0.032, +0.008]
$(\mathbb{Z}_2)^6$	0	8	128	-0.1925	-0.1585	-0.0340 [-0.063, -0.005]
$(\mathbb{Z}_2)^6$	1	8	128	-0.3175	-0.3069	-0.0107 [-0.053, +0.032]
$(\mathbb{Z}_2)^6$	2	–	–	–	–	Capability failure
$(\mathbb{Z}_2)^6$	3	6	126	-0.1436	-0.1796	+0.0361 [-0.016, +0.088]
$(\mathbb{Z}_2)^6$	4	10	128	-0.1741	-0.2530	+0.0789 [+0.035, +0.123]
D_{31}	0	11	128	-0.0586	-0.2380	+0.1794 [+0.138, +0.221]
D_{31}	1	10	128	-0.1398	-0.1929	+0.0530 [+0.008, +0.098]
D_{31}	2	–	–	–	–	Capability failure
D_{31}	3	–	–	–	–	Capability failure
D_{31}	4	10	128	-0.0135	-0.1942	+0.1807 [+0.139, +0.222]
A_5	0	11	128	-0.1125	-0.2382	+0.1257 [+0.104, +0.147]
A_5	1	10	128	+0.0830	-0.1722	+0.2551 [+0.213, +0.297]
A_5	2	11	128	+0.0943	-0.0736	+0.1680 [+0.125, +0.211]
A_5	3	11	128	+0.0799	-0.2116	+0.2916 [+0.205, +0.378]
A_5	4	11	128	+0.0917	-0.1656	+0.2573 [+0.210, +0.305]

The original all-sixteen sign criterion is not met. Missing capability and opposite-sign measurements remain explicit. The previous raw-logit gate failures are numerical-infrastructure results, not evidence that the eventual A2 contrasts are unstable. The repair is a documented change to the measured consistency target; it does not erase the original failure record.

Axis and amplitude audit. The archived simultaneous array has axes $(\text{task/null/delta}, \text{start}, k)$ and is computed at .001 only. Its two starts are the independent calibration peak and t_{80}, ordered by update. Reading them as two radii is invalid. We rebuild a four-axis array from every raw context, $(\text{task/null/delta}, \text{start}, k, \epsilon)$, for .001 and .003. Both actions and directions, all starts and all four windows define the common mask. It exactly reproduces every archived mean and context count, and the two radii have identical eligibility masks. There are 2,046 included contexts: fifteen sets of 128 and one of 126. No sign flips occur in the primary task, null or contrast means under the actual radius comparison. The maximum primary task-log-gain change is 0.000200 at one update and 0.000732 at four; contrast changes are smaller. This is a finite-radius consistency test over specified cells, not a bound on every individual direction or larger perturbation.

Calibration association. Mean calibration accuracy is the arithmetic mean over the 28 recorded depths. Spearman uses average ranks for ties. The reported p values use 200,000 random permutations and a plus-one correction, with seed 20260922. The within-group statistic centers both raw variables within group and permutes within groups. These post hoc associations are reported descriptively without treating nominal significance across the examined contrasts as confirmatory. They do not establish collinearity, mediate the group effect, or demonstrate a zero conditional effect. They identify inference progress as an unresolved alternative to a causal algebraic explanation.

R Normalization and the readout-norm test

With output-side state $z = C_c(h)$, the relevant readout is

$$\ell_c(h) = s\, W R_\gamma(C_c(h)), \qquad R_\gamma(z) = \frac{\gamma \odot z}{\sqrt{\|z\|_2^2/d + \eta}}, \tag{11}$$

where s is a logit scale, C_c includes exit processing and γ is trainable. For $b > 0$ and $\eta = 0$, $R_\gamma(bz) = R_\gamma(z)$ (Zhang & Sennrich, 2019). For $\eta > 0$, the equality is approximate when the squared RMS dominates η. Orientation and the learned gain remain consequential. The effective differential is

$$D\ell_c(h) = s\, W\, DR_\gamma(C_c(h))\, DC_c(h).$$

Thus radial state contraction need not be compensated by an increase in $\|W\|$, and $\|W\|$ does not determine response to a task direction. Tying sets $W = E$, the input embedding matrix. It combines input and output gradient contributions; it does not set $\nabla W = 0$ or constrain $\|W\|$ to a fixed value. Exact storage equality in twenty saved checkpoints and the range 10.94–56.62 directly disprove a frozen-norm claim.

The earlier CPU extraction is retained as an executed diagnostic. Within groups of new capable seeds, exact one-sided Spearman tests of a negative association between task gain and full-head Frobenius norm have (ρ, p) equal to $(-1, 1/6)$ for Z61, $(-0.5, 0.5)$ for XOR and $(-0.4, 0.375)$ for A5. D31 has only two values and is descriptive. None passes four-group correction; with at most four seeds, even perfect ranks cannot attain that threshold. These outcomes do not prove compensation impossible. They show that this norm-only test neither identifies nor establishes it.

S Finite-grid boundary definitions and controls

For a 513×513 label array Q_T with V possible tokens, the uncertainty statistic at integer offset e is

$$f_T(e) = \frac{1}{2}\left[\frac{\sum_{i,j} \mathbf{1}\{Q_T(i,j) \neq Q_T(i,j+e)\}}{513(513-e)} + \frac{\sum_{i,j} \mathbf{1}\{Q_T(i,j) \neq Q_T(i+e,j)\}}{513(513-e)}\right],$$

where each sum is restricted to valid endpoint pairs. It is an average of two axis-aligned pair-disagreement fractions. OLS of $\log f$ against $\log e$ on $e = 1, 2, 3, 4, 5, 6, 8$ defines α_{eff}, a finite-grid analogue of the uncertainty exponent (McDonald et al., 1985). At 560 every disagreement count is zero, so no logarithmic fit exists. We report undefined, not a dimension of zero.

For mixed-box counts, use non-overlapping boxes anchored at the origin, with sides 2, 3, 4, 6, 8, 12, 16, 24, 32, 48 and 64 pixels. Crop the bottom and right remainders. Count a box when it contains multiple labels. The negative slope of log count against log side defines $d_{\text{box}}^{\text{eff}}$. This is a

different estimator, but its observations are not independent of the uncertainty statistic. Scale points are few and dependent; residual OLS errors do not include variation across grid origins, checkpoints or prompts, or uncertainty about the continuum limit.

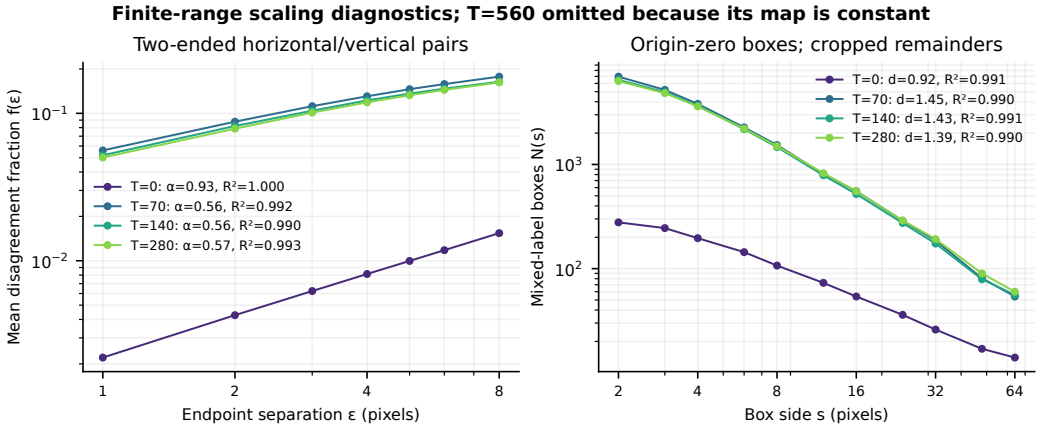


Figure 17: **Finite-range slopes and controls.** The same saved token grids underlie both estimators. Large slopes and high regression fit quality do not suffice to identify an asymptotic fractal boundary: known smooth periodic partitions can inflate mixed-box slopes. Changing the uncertainty fit range changes intermediate-map slopes. All controls and per-scale counts are available in the independent boundary JSON.

Matched smooth controls give uncertainty effective dimensions near one. A five-pixel checkerboard gives approximately 1.459 but a poor uncertainty-fit R^2 of 0.473; an eight-pixel checkerboard gives approximately 1.000 and excellent fit, yet its mixed-box estimate is approximately 1.785. A three-pixel checkerboard gives a mixed-box estimate near 1.916 with high fit quality. Therefore neither a large exponent nor two apparently concordant estimators is a stand-alone fractality test. The intermediate data's good uncertainty fits apply to the chosen 0.90-decade range; range dependence prevents an asymptotic claim.

The T=0 panel is an empirical control budget, not an analytically proven smooth partition: its uncertainty estimate is 1.067, not exactly one. Token IDs are assigned a common categorical palette across panels. The source NPZ stores row-major point IDs, five checkpoints and 263,169 evaluations; a second archive has exactly matching tokens/checkpoints. The separate last-switch field retains 332 values at T=560. A map of last-switch times and a next-token partition answer different questions.

T IDEAL SYMMETRY TRANSPORT AS A REFERENCE IDENTITY

For a group representation ρ and fixed prompt c, if

$$F_c(\rho(g)h) = \rho(g)F_c(h),$$

then induction gives $F_c^t(\rho(g)h) = \rho(g)F_c^t(h)$ for every integer $t \geq 0$. If the transformation also changes the prompt, the appropriate relation is instead

$$F_{gc}(\rho(g)h) = \rho(g)F_c(h), \qquad F_{gc}^t(\rho(g)h) = \rho(g)F_c^t(h).$$

These are exact statements under the displayed assumptions. A2 holds the recipient prompt fixed after defining its directions, so it is not a direct test of the second identity. We do not identify a latent action ρ or measure its commutation defect here.

Equivariance alone imposes no universal gain sign. Multiplication by a scalar commutes with every linear group action, regardless of whether its magnitude is above or below one. With multiple invariant subspaces, blockwise gains can differ while preserving equivariance. All finite-group representations over the reals are completely reducible; that property does not separate these four tasks.

The XOR group has only one-dimensional real irreducibles, but a faithful representation of $(\mathbb{Z}_2)^6$ needs multiple such components, not a single faithful one-dimensional representation. No algebra-to-sensitivity-sign theorem follows from irreducible dimension alone.

U ARTIFACT AND VERIFICATION SCOPE

Experiment I (independent statistical constructions), Experiment II (matched donor congruence), and the earlier discovery studies have separate plans, source hashes, checkpoint manifests and result archives. Both principal experiments froze inputs, code, numerical gates and inferential families before measurement. The four failed training replicas and the retired technical-pilot context remain recorded. No failed replica was silently replaced.

The complete archives contain seven conditions over 8,192 pairs for Experiment I and six conditions over 16,384 pairs for Experiment II, with full logits at lags 0–35, capped endpoints and event flags. Independent array-based implementations reconstruct predicted labels, endpoints, block aggregates and simultaneous intervals. File and archive-member hashes check transfer integrity. These checks establish arithmetic and provenance under the stated design; they are not human replications or finite-sample coverage proofs.

Post-hoc families remain separate from registered ones. Experiment I's fixed-mean timing/noise, native–donor and repeated-output analyses are described in Appendix A. The new Experiment II insertion audit and 24-cell sham families are described in Appendix C, and the saved-logit analyses of stopping rules, wrong labels and the certificate in Appendix D; neither retroactively alters the registered criteria. Historical four-checkpoint certificate and baseline audits use their own archived trajectories and are not treated as sixteen-checkpoint replications.

The reproducibility bundle records neutral study identifiers, source-to-archive mappings, scripts, per-pair values and build provenance. Numerical outputs were recomputed by independent scripts to the scopes stated above. No figure changes measurement values or inferential thresholds. Authors retain responsibility for scientific interpretation and the submitted manuscript.

V A MEASURED COUNTDOWN STATE SLICE

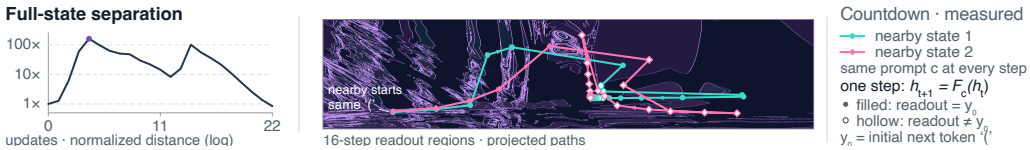


Figure 18: **A measured state slice of a separate Countdown model** [X]. Left: full-state distance of two neighboring initializations relative to display start (log scale). Right: projected trajectories over 16-update readout regions; hollow markers indicate departure from the shared initial token. Neighboring initializations, not a donor/sham pair; details below.

Model and scope. This exploratory illustration uses the Countdown Parcae checkpoint and fixed prompt of Figure 6, separately from the finite-group intervention experiments. Each recurrent state has $15 \times 768 = 11{,}520$ coordinates. The background and foreground use the same checkpoint, continuing input and affine state plane. The background is a finite-update next-token partition; it is not an inferred vector field or an asymptotic basin classification.

Two nearby states with the same readout. We selected neighboring initializations from the archived 513^2 plane that shared their initial readout but differed after 70 updates. Candidate pairs were ranked by their smaller 70-update margin within spatial bins; all candidates and the selection rule are saved. The displayed pair is the first candidate, at original coordinates $(-2.68359375, -1.76953125)$ and $(-2.68359375, -1.7578125)$ in units of the archived plane radius. We replayed both through 560 updates, saving full states. The slice starts at global update 10, where both decode token 10, (, and ends at update 32. Thus it shows 23 sampled states connected by 22 update segments. Neither trajectory is an intact recipient or a transplanted donor. For each measured update, filled markers denote readout $y_0 =$ (and hollow markers denote a different next token. State 1 is hollow at local updates 6, 7 and 14–22; state 2 is hollow at 5–22. This indicates departure from the initial token, not an incorrect final answer. Each segment applies $h_{t+1} = F_c(h_t)$ under the same prompt c. The starting separation is 124.434, or 1.934% of the first state's norm. Shared readout here means the same next token, not a shared correct final Countdown solution. The left inset is recomputed from these same full states: $g_k = \left\| h^{(2)}_{10+k} - h^{(1)}_{10+k} \right\|_2 / \left\| h^{(2)}_{10} - h^{(1)}_{10} \right\|_2$, $k = 0, \ldots, 22$. Its logarithmic ordinate reports distance relative to display start, independent of the projection. The selected pair reaches $160.85\times$ at $k = 4$ (global update 14), and ends at $0.843\times$. The pair, window and projection are selected for illustration, not a confirmatory estimate of typical trajectory divergence.

One plane for the map and both trajectories. Let $c = h^{(1)}_{10}$ and $e_2 = (h^{(2)}_{10} - c) / \left\| h^{(2)}_{10} - c \right\|_2$. After removing the e_2 component from all displacements $h^{(i)}_t - c$, $i \in \{1, 2\}$, $t = 10, \ldots, 70$, we take their leading right singular vector as e_1. The resulting orthonormal basis defines

$$\mathcal{P} = \{c + xe_1 + ye_2 : (x, y) \in \mathbb{R}^2\}, \qquad \pi(h) = \big(e_1^\top (h - c), e_2^\top (h - c)\big).$$

Both displayed starting states belong to $\mathcal{P}$ up to rounding; their initial distance is preserved. Subsequent full states generally leave this plane: their orthogonal projections, rather than trajectories constrained to $\mathcal{P}$, are drawn. Straight segments connect discrete updates without temporal smoothing. The rectangular display scales its two axes independently to fit the fixed horizontal panel; screen angles and distances are therefore not the original Euclidean metric.

A directly measured high-resolution background. We sample $1{,}026 \times 1{,}026 = 1{,}052{,}676$ initial states $c + xe_1 + ye_2$ on the common plane. The bounds are the extrema of the two displayed projected trajectories, extended by 20% of their span on each side: $x \in [-2221.931, 13331.584]$ and $y \in [-2653.582, 14790.259]$. Every sampled state receives 16 applications of the same prompt-conditioned map. The raster records the next-token argmax after those 16 updates. Dark purple shades distinguish readout regions, and light purple marks adjacent grid points with different readouts. This restores the original boundary rendering; colors do not encode switching time. This is

a fresh scan at the displayed resolution, not an upsampled pilot. The foreground is not advanced according to this background, which describes continuations from the plane rather than the states' instantaneous motion. Fine finite-resolution boundaries alone establish neither a fractal limit nor chaos.

Numerical audit and retained failure. All evaluations use FP32, disabled TF32 and math attention; plane states are constructed in FP64 and cast once to FP32. The original all-time singleton-versus-batch relative-state threshold 2×10^{-5} failed (maximum 3.30×10^{-3}). This failure is retained, and the illustration does not claim that gate passed. In the separately examined display window, the actual pair separation exceeds the sum of both numerical discrepancies by at least $133.3\times$; the discrepancy in the final rectangular screen coordinates is at most 0.076 pt, against a 0.76 pt trajectory stroke. Singleton and paired replays agree in next-token argmax at every saved update. These checks resolve the displayed paths at their plotting scale, not exact state invariance or arbitrarily fine boundaries.

Reproducibility. The supplementary bundle retains the full-state replay, candidate list, projection, failed-gate record, pilot and high-resolution maps, and the recomputed separation used by the inset. The high-resolution plan, code, plane and input hashes were frozen before the scan. Raw pointwise outputs, finite-value flags, input identities and chunk hashes are retained, and automated checks verify complete grid coverage, finite outputs and transferred hashes.

W A FROZEN TRANSPLANT TEST IN A COUNTDOWN MODEL

[R] The main text studies looped transformers trained on finite-group composition. To see whether the effect of Section 3.1 appears in a different task and in a checkpoint trained by other researchers, we ran one frozen test on the Parcae Countdown checkpoint of Figure 6 (Prairie et al., 2026; setup of Lai et al., 2026). Design, code and decision rule were frozen by hash before measurement (pre-registration SHA-256 prefix `9237663a`; not externally registered). The authors fixed the equivalence margin, the minimum sample and the problem count before any transplant continuation existed. This is the only contrast we test in this model; everything else in this appendix is descriptive.

Design. We draw 2,048 new problems with five numbers in 1–20 and a reachable target in 1–99, so every prompt has the same token length. The readout is the next token after the prompt (the first token of the expression), read at the last prompt position after every update. Because Countdown has many correct expressions, the reference is not a correct answer but R_1, the recipient's own untouched readout at update 27. A *deviation* is a readout that differs from R_1. Unlike an *error* in the main text, it is not checked against a solution, and rates must be compared with this in mind. The answer-preserving donor c^+ swaps the first two numbers (token positions 1 and 3); the numbers and the target, and hence the set of solutions, stay the same, as with the matched donors of Eq. (1). Each donor runs 200 updates under its own prompt from the recipient's initial state, and its state at the insertion update replaces the recipient's. We keep a donor as *answer-preserving at the model level* (type i) when its own untouched run settles on R_1 (a constant readout over updates 150–200); 1,395 of 1,945 swap donors qualify. States are inserted at update 7 (the first of three consecutive updates at which at least 80% of calibration problems read R_1, the analogue of the insertion time of Section 2) and, as a secondary time, at update 27. The test set P holds the recipients whose type-i transplant and untouched run both read R_1 at lag 0. The outcome E_8 is the share of P with a deviation at any of lags 1–8, and the registered contrast is $\Delta E_8 = E_8(c^+) - E_8(\text{untouched})$, paired by recipient, with a 95% percentile interval over 50,000 bootstrap draws of problems. The rule, fixed in advance, reads the contrast as positive if the lower limit exceeds zero and as equivalent if the interval lies within ± 3 points; it requires at least 400 pairs. A positive result must be reported with two qualifiers: whether the difference remains within strata of the lag-0 readout margin (top-two logit gap), and the contrast in the stratum where R_1 is not the value of either swapped number, since a model that points at one of those numbers would deviate for a simple reason.

Frozen verdict. P holds 1,066 pairs. The untouched run deviates in 2.06% of them and the transplant in 3.75%, so $\Delta E_8 = +1.69$ points [0.58, 2.81]. By the frozen rule the contrast is *positive*: in this Countdown model too, answer-preserving transplants that read the recipient's answer at insertion deviate within eight updates more often than the untouched run. The registered qualifiers are part of the verdict. (i) The interval also lies within the ± 3-point margin (the rule gives a positive result precedence over equivalence), so the difference is smaller than that margin. (ii) Within quartiles of the lag-0 margin the differences are 0.37–1.37 points, and their average over quartiles is $+0.86$ $[-0.25, 1.96]$; the difference does not remain within margin strata, so it can be explained by the readout margin at insertion. (iii) Where R_1 is not the value of either swapped number, $\Delta E_8 = +1.33$ $[-1.04, 3.81]$ over 301 pairs (where it is: $+1.83$ [0.66, 3.06], 765 pairs). This interval includes zero, so a simple pointer mechanism can explain the positive result, and the result is weak as support. Re-inserting the recipient's own state (the sham) reproduces its uninterrupted readouts at every lag 0–8 in P, no readout in P is non-finite at lags 0–8, and by construction the sham reads R_1 at lag 20 (update 27), outside the E_8 window. The four registered sensitivity analyses (only recipients whose update-27 and update-200 answers agree; type i judged at update 27; near-tie deviations ignored; numerical mismatches excluded) all give $+1.69$ points with limits within 0.01 points of the primary ones. A second implementation inside the frozen analysis script, also written by Claude, reproduces every checked count.

Descriptive results. Intervals for the registered secondary family (answer-changing donors, random displacement, margin strata, later insertion) are simultaneous 95% intervals; the others are unadjusted percentile intervals. Donors that settle on another answer but whose transplants read R_1 at lag 0 (type ii, 208 pairs) deviate much more often (17.31% against 6.73%, $+10.58$ [5.94, 15.52]); a design review had anticipated this selection effect, which is why they are outside P. Without any filter at update 7, the untouched run reads R_1 at lag 0 in 85.7% of the 1,945 recipients with a swap

donor and the swap transplant in 70.6%, and they deviate within lags 1–8 in 13.6% and 27.2%. Keeping only type-i donors shrinks the difference to $+5.23$ points [3.44, 7.05], and adding the lag-0 filter of P gives the frozen $+1.69$. Inserted at update 27, the contrast is $+2.59$ [1.29, 3.89] over 1,198 pairs, again small. A random displacement of the same norm at every position deviates more than the swap transplant ($+4.92$ [2.17, 7.68], 731 pairs). Swapping the last two numbers instead of the first two leaves the median first deviation at lag 2 (difference 0, interval [0, 1]), as expected if a small difference reaches the last position within one update (Appendix G.10, the Countdown probe). Transplants from answer-changing donors (as the swap, but with a new number at position 1, kept when their update-27 answer differs from R_1) settle on R_1 0.34 updates later than swap transplants [0.20, 0.48], and their eight-repeat exits stop off R_1 about as often ($+0.48$ points [-0.34, 1.30], 1,869 recipients). At lag 35, 99.59–99.85% of continuations read R_1 in every condition.

Relation to the main text. In the finite-group checkpoints at length 8, 53.36% of answer-preserving transplants that start correct err within eight updates, against 0.84% of untouched runs (Section 3.1). In this Countdown model the corresponding difference is more than an order of magnitude smaller than that pooled one, its upper limit lies below the difference in every finite-group checkpoint, and it can be explained by the readout margin. The effect's size thus does not carry over to this Countdown model, and the main text states its claims for the finite-group checkpoints only. The models differ in task, training and readout (deviation vs. error), so the comparison cannot say which difference matters, nor can one checkpoint show the effect absent elsewhere.

Changes made before freezing. Two specification–code differences were found before the measurement existed; the frozen plan and code implement the versions below. The analysis first judged type i by the donor's update-27 answer instead of its settled answer; both scripts were corrected, and the update-27 version became a sensitivity analysis (above). A design draft compared the swap transplant with a state computed from another initial state; the frozen plan replaced that comparison by the norm-matched random displacement reported above. The problem count was raised from 1,024 to 2,048 before freezing, when no transplant continuation existed.