LEARNING TO REPAIR THE LEARNER: EVALUATING RECURSIVE IMPROVEMENT IN NEURO-SYMBOLIC WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-improving agents are usually credited through descendant-performance curves: later lineage members outscore their ancestors. Such curves misattribute improvement even in a fully inspectable system. We formalize a checkpoint separating world model, repair procedure, evidence and a fixed kernel, and derive a ladder of matched interventions, from crossed model/procedure states and owned versus replayed evidence to a code$\times$history crossing, a generator lock and non-recursive baselines. We apply it to a neuro-symbolic lineage that edits its repair procedure and successor generator, on an original cohort and replications with 252 fresh families declared before execution. In replication, inherited procedure explains 45% of the naive descendant gain; model structure explains the rest. Under a shared source history, continued editing detectably improves on freezing, but through final code (history effect ≈ 0), and is practically equivalent to choosing one fixed behavior per family on the same continuation evidence with fewer fits; generator editing is practically equivalent to locking it. A positive control shows the ladder detects planted generator and history value. Because the base generator indexes proposals by history length, sharing a history between source lineages decides whether checkpoints lock into single-term repair (504/504 versus 75/504); with separate histories the gain over freezing is no longer detected. A replicated gain thus hinges on the checkpoint's history contract, the failure our formalism exists to expose.

1 INTRODUCTION

Claims that an agent improves itself are usually supported by descendant-performance curves: a later member of a self-editing lineage outscores its ancestor. Such a curve cannot say *why*. The descendant may inherit a more accurate model, own more informative evidence, search more candidates, or benefit from what its checkpoint lets it share with other lineages. None of these is improvement of the procedure that does the improving, yet all of them produce the same curve. Separating them requires matched interventions, not better curves.

We make this separation operational for agents that repair executable world models. A checkpoint explicitly separates the world model, the repair procedure (diagnosis, model repair and successor generation), accessible evidence and a fixed evaluation kernel. Each claim about self-improvement is then tied to one matched intervention, forming an evidence ladder (Table 1). We apply the ladder to a neuro-symbolic lineage that edits its own repair procedure *and* the generator that proposes its successors. The testbed is small by design: every control is executable, every procedure is inspectable, and a full family runs in about 1.5 s, so the controls can be run on hundreds of independent world families and crossed exhaustively. In it, a continued-editing gain that survived replication turned out to hinge on an incidental checkpoint detail: how long a history two source lineages share.

Findings. Each rung qualifies the verdict that the descendant curve suggests; numbers are from the $n = 252$ replications unless stated otherwise. (i) *Inherited structure carries more of the gain than inherited procedure.* The naive descendant gain -0.00895 ($[-0.01962, -0.00248]$) splits into procedure and structure main effects with point-estimate shares 45% and 55%. (ii) *The evidence regime changes the estimate.* Donor replay overstates the inherited procedure's benefit relative to

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.

owned evidence (0.00987, $[0.00036, 0.02570]$). (iii) *Under a shared source history, continued editing detectably improves on freezing, and the gain is final code.* Crossing final code with final history, the code effect nearly equals the package effect and the history effect is exactly zero in 247 of 252 families. (iv) *Evidence-matched non-recursive selection matches the lineage.* Choosing one of the six evaluation-time behaviors per family on the lineage's own continuation evidence, with 144 rather than 192 fits, is practically equivalent to continued editing (continued minus baseline 0.00142, 90% interval $[-0.00029, 0.00329]$ inside ± 0.005). (v) *Generator plasticity adds nothing detectable*: continued minus generator-locked is 0.00031 (90% $[-0.00171, 0.00219]$). (vi) *The ladder can return positive verdicts.* When generator and history execute at test time, the same contrasts detect planted effects (Section 5.3). (vii) *The gain over freezing is fragile.* A pre-declared test rules out a curriculum explanation. Instead, the base generator indexes its proposals by history length, so whether the inherited and fixed-meta source lineages share a history decides whether the checkpoint locks into single-term repair; with separate histories the continued-over-frozen gain is no longer detected (-0.00121, $[-0.00316, 0.00121]$), while non-recursive per-task validation selection is detectably better than continued editing in both regimes (Section 5.4). In short: most of the naive gain is inherited model support, and the continued-editing gain is largely one repair-subtree flip that non-recursive selection also finds and that an incidental design detail makes necessary.

Contributions.

- (C1) A checkpoint formalism and evidence ladder that separate model repair, inherited repairer capability, evidence-regime effects, executed recursion, the value of continued editing, and the value of generator plasticity, each tied to a specific matched intervention (Section 3).
- (C2) An inspectable neuro-symbolic testbed, released with all controls, in which each intervention is cheap enough for tests on 252 families declared before execution, equivalence tests against a declared margin, exact code$\times$history crossings, and a positive control that plants known generator and history value (Sections 4–5).
- (C3) A case study in which the ladder overturns the naive reading: apparent recursive self-improvement is largely inherited structure and is inflated by replayed evidence; the continued-editing gain is a final-code effect, practically equivalent to non-recursive selection on the same evidence, and hinges on an incidental feature of the history contract between source lineages; generator editing is practically equivalent to locking it (Section 5).

2 Related work

Editing the improvement machinery. STOP recursively applies an improver to its own code, and Hyperagents makes task and meta agents editable and evaluates later improvement ability (Zelikman et al., 2024; Zhang et al., 2026a). DGM explores self-modifying agents, AEvo edits the evolution mechanism, and self-modifying Lean agents rely on an external proof verifier (Zhang et al., 2025; 2026b; Li et al., 2026); Dream-RSI evolves exploration-policy code under a fixed development agent (Zheng et al., 2026). These systems establish that improvement machinery can be edited; they credit it mainly through descendant performance, in some cases with frozen-meta or budget-matched comparators (Appendix G). We ask which part of such a gain survives crossing model and procedure, crossing code and history, locking the generator, changing the history contract, and comparison with non-recursive search.

World-model repair and abstractions. WorldCoder learns executable transition and reward programs, PoE-World composes programmatic experts, WorldCloner adapts symbolic dynamics to novelty, and VisualPredicator learns grounded predicates and abstract actions (Tang et al., 2024; Piriyakulkij et al., 2025; Balloch et al., 2023; Liang et al., 2025). CoEx couples modeling with exploration, and DreamCoder builds abstractions that alter later search (Kim & won Hwang, 2025; Ellis et al., 2020); neural world-model planning is a strong neighboring class (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2021; Hansen et al., 2024). These methods improve the model or library; our object is the procedure that repairs it.

Diagnosis and controlled evaluation. SkillMentor learns diagnosis separately from a frozen executor, and AutoMem uses failure attribution to direct architecture search (Bao et al., 2026; Du

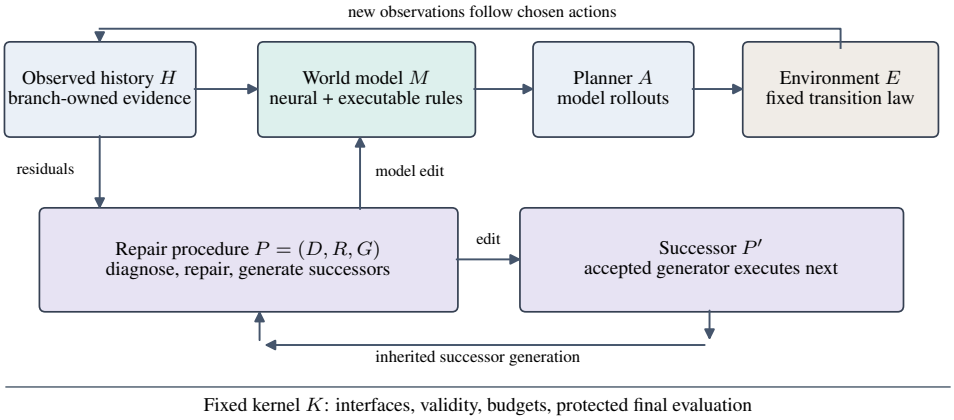

Figure 1: Objects and information paths. World-model repair and procedure editing have separate identities. The fixed kernel defines interfaces, validity and budgets; protected final evaluation supplies no search feedback. An accepted successor must execute before its inherited generator is credited with later proposals.

et al., 2026). ANNEAL validates typed symbolic patches, HELIX records harness interventions and provenance, and CHILL-Harness uses checkpointed paired counterfactual execution with matched continuation and resource accounting (Hakim et al., 2026; Fan & Huang, 2026; Fu et al., 2026); diagnosis of planning failures and reflective learning have a long history (Birnbaum et al., 1990; Cox & Ram, 1999; Fox, 1998). Checkpoints and counterfactual branches therefore have direct precedents; Appendix G maps the broader comparison class from documented interfaces only. Our contribution is not any single control (model/procedure crossings, frozen baselines and generator locks appear in DGM-, Hyperagent- and CHILL-style work) but a checkpoint formalism that specifies what must survive between generations, and a jointly applied ladder, matched on evidence and fit budget, whose replication, curriculum, positive-control and lineage tests were declared before execution. Applied to our own system, it reversed our own conclusions twice (curriculum, shared history).

3 What a self-improvement claim requires

Let a checkpoint be $X = (M, P, H, L, E, \omega; K)$: world model M, repair procedure P, accessible history H, reusable library L, environment state E, random state ω, and fixed kernel K. Write $P = (D, R, G)$ for diagnosis, model-repair generation and procedure-successor generation. A model edit changes M; a procedural edit changes executed behavior in D, R or G. Recursion requires more than a changed source hash: an accepted change to G must subsequently execute and participate in producing another procedure. Figure 1 shows the information paths; Table 1 lists the required evidence.

Frozen-procedure capability. Let $Y(M, P; \mathcal{E}, B)$ be lower-is-better final loss after a frozen procedure repairs M using supplied evidence $\mathcal{E}$ and budget B, and write $Y_{ij} = Y(M_i, P_j; \mathcal{E}, B)$. The procedure main effect and interaction are

$$\Delta_P = \tfrac{1}{2}[(Y_{01} - Y_{00}) + (Y_{11} - Y_{10})], \qquad I = Y_{11} - Y_{10} - Y_{01} + Y_{00}, \tag{1}$$

and the structure main effect averages $Y_{10} - Y_{00}$ and $Y_{11} - Y_{01}$. The naive descendant gain is the diagonal $Y_{11} - Y_{00}$, which equals the sum of both main effects; Equation 1 apportions it. The supplied history must be compatible with every cell, and a predictor-dependent diagnostic is recomputed when its predictor changes.

Adaptive future value. Let $\mathrm{Run}(X, P, B)$ allow a procedure to choose probes and receive their outcomes. A shared-start contrast is

$$\Delta_{\mathrm{adapt}} = \mathbb{E}[Y(\mathrm{Run}(X_0, P_1, B)) - Y(\mathrm{Run}(X_0, P_0, B))]. \tag{2}$$

Claim	Required comparison	Insufficient observation
Model repair helps	Repaired versus original model on protected outcomes.	Improved development fit.
Inherited repairer helps	Frozen procedures on matched model/evidence states.	Better descendants with more history.
Evidence is transportable	Owned versus donor-replayed evidence from a shared start.	A replay-only comparison.
Recursion executes	Changed generator participates in a later proposal.	Different serialization.
Continued editing helps	Editable versus frozen from one checkpoint; final code crossed with final history.	An initial-to-final gain.
Generator edits help	Editable versus generator-locked procedure.	Beating a fully frozen repairer.
Recursion is necessary	Improves on non-recursive search over the same behaviors and evidence (fixed meta, validation-selected, evidence-matched).	Beating only an untuned ancestor.
Contract is neutral	Result survives changing what lineages share (one versus separate histories).	One checkpoint contract.
Ladder has power	Planted effects are detected when the component executes.	Nulls alone.

Table 1: The evidence ladder. Each row asks a separate question; passing an earlier row does not establish a later one.

Branches begin with equal state and evidence, then own their observations. Replaying a donor's fixed tuples instead estimates utility conditional on that donor's evidence: two environments can agree on every donor tuple yet assign opposite outcomes to an action the donor never took (Appendix F). Logged-feedback estimators need support and identification assumptions that replay does not supply (Dudík et al., 2011; Jiang & Li, 2016; Swaminathan & Joachims, 2015). The adaptive-minus-replay difference is an evidence-regime interaction, not a mediation estimate.

Value of continued editing. From the same inherited checkpoint X_k, continuation with G_k enabled is compared with a branch that keeps P_k frozen; both still receive the same new evidence and repair their models. Because the resulting endpoint evaluates a program together with its accumulated history, the code and history of the final checkpoints must be crossed to credit the code. A generator-locked branch permits every other procedure edit and isolates generator plasticity. Finally, recursion is only necessary if it beats non-recursive search over the same evaluation-time behaviors: a fixed history-aware meta-procedure, per-task selection by validation error, and one behavior per instance chosen on the lineage's own continuation evidence.

Checkpoint and resource contracts. A checkpoint is an information contract as well as a serialization. Transferring a repairer's code without its residual cache or library changes the treatment; transferring every dependency can give one branch extra evidence; letting two lineages write into one history couples their searches. Each comparison therefore states its object (code alone, code plus diagnostic state, or the deployable package) and either equalizes compatible state or reports it as part of the treatment. Repeated selection risks overfitting the development evaluator (Cawley & Talbot, 2010; Smith & Winkler, 2006; Dwork et al., 2015), so the fixed kernel owns the final test and a ledger of observations, fits, proposals and rejected edits. Independent world families, not correlated descendants, are the unit of uncertainty.

4 Executable reference mechanism

The reference environment is a two-state, one-dimensional controlled dynamical system. Position x, velocity v, context c and action u determine the next state. The world model combines a context-dependent action term with a selected set of symbolic force terms:

$$\widehat{v}' = .52v - .10x + u\,\theta^\top \phi(c) + \textstyle\sum_{j=1}^{4} z_j \beta_j \psi_j(x, v, u), \qquad \widehat{x}' = \mathrm{clip}(x + .35\widehat{v}', -2, 2). \tag{3}$$

Here z_j specifies symbolic support, ϕ contains a constant and three fixed tanh features, and ψ contains drag, cubic restoring, action-saturation and one-sided restoring terms. A ridge solve jointly

fits the neural output coefficients θ and active symbolic coefficients β. The simulator uses the same functional family with independently sampled coefficients. This deliberately realizable setting isolates procedural behavior from representation learning (Appendix A).

Procedure grammar. A procedure is executable code assembled from four function subtrees. Diagnosis orders symbolic terms cyclically, by residual correlation, or by residual correlation adjusted with historical edit gains. Repair proposes single-term or paired-term edits. Acquisition uses passive actions, model disagreement, or disagreement with a goal penalty. The successor generator uses cyclic enumeration, empirical edit credit, or composition of two subtree changes. The product contains 54 programs; a legal successor changes one or two subtrees. The interpreter, primitive dictionary, ridge solver, promotion rule and planner are fixed. Source code is emitted and executed, but the grammar makes this a finite search problem rather than open program synthesis.

Development and promotion. A candidate is scored on two consecutive repair tasks: it repairs the first model, its generator emits a descendant, and that descendant repairs the second. The mean validation error is its procedural credit, so generator changes affect selection. Four candidates and the incumbent receive identical budgets, and promotion requires strict validation improvement. The fixed meta-procedure searches the same legal edits using edit-class credit and credit keyed by the diagnosis/repair pair, with its own fixed generator in the lookahead. Proposal functions do not read mechanism labels or planning outcomes. Acquisition is not scored in the development objective, so inherited acquisition behavior is incidental. During source training the inherited and the fixed-meta source lineages see the same stage tasks and write their candidate records into *one shared history*. The base program's cyclic generator starts its proposal window at the history length modulo the number of legal edits, so this contract, used in every cohort except Section 5.4, changes which edits the inherited lineage is offered.

Evaluation and the six-behavior reduction. An exhaustive receding-horizon planner selects among five actions using three-step predicted costs; protected evaluation reports state-tracking cost (the primary planning loss) and transition prediction error. Each final task starts from empty model support and fresh common evidence, and the frozen final procedure repairs the model. During this repair only the diagnosis and repair subtrees execute: acquisition is not called and the generator proposes nothing. The 54 programs therefore reduce to $3 \times 2 = 6$ evaluation-time behaviors, and a program's history enters only through history-adjusted diagnosis. This structural fact, which follows from the repair interface and which the crossing script checks on the first four families of each cohort, makes an exhaustive non-recursive baseline possible: every behavior can be scored on every task with every final history.

5 Experiments

5.1 Design, inference and registration

Each world family is an independently seeded draw of simulator coefficients. Four source stages expose single symbolic mechanisms, four continuation stages combine pairs, and final transfer tests the four combinations of three mechanisms, so transfer withholds compositions rather than primitive physics. Every family is paired across all conditions and contributes one task-averaged contrast; we report paired means with percentile bootstrap 95% intervals over families (10,000 resamples). The original cohorts ($n = 32$ each for E1–E3 and for the generator-lock follow-up) are discovery cohorts; the replications ($n = 252$ each, fresh seeds, unchanged code) carry every claim and are never pooled with them. Table 2 states once what was declared before execution.

The equivalence margin $\delta = 0.005$ (about 0.55% of the original cohort's mean transfer loss and one third of the original continued-versus-frozen benefit) was fixed after the original results and before the replications, and it is applied to every contrast. A contrast is *detected* if its 95% interval excludes zero, *practically equivalent* if its 90% interval lies inside $\pm\delta$ (two one-sided tests), and *practically relevant* if its 90% interval lies outside $\pm\delta$; a contrast can be detected but practically equivalent. Holm adjustment (Wilcoxon) over each cohort's four pre-declared primaries keeps all four in the replication but only continued minus frozen in the original cohort (Holm $p = 1.2 \times 10^{-4}$). There, one family carries 77% of the E1 procedure sum and 51% of the E2 interaction, and without it neither

Experiment (n families)	Status	Declaration
E1–E3; generator-lock follow-up (32 each)	pre-declared	Frozen before execution; follow-up designed after main results were opened.
Replications (252 each); misspecification (252)	pre-declared	Hashed before any seed ran, after the original results; δ fixed here; misspecification (no declared primaries) re-frozen once before any seed ran.
Curriculum intervention (252); lineages (126×4)	pre-declared	Hashed with predictions; one post-freeze amendment to a descriptive lifetime-cost quantity.
Positive control (252)	pre-declared	Hashed; 16 pilot seeds inspected before predictions were written.
Code×history crossing; behavior and evidence-matched baselines; Holm/BCa/influence; lifetime cost; edit traces	exploratory	Post-hoc, undeclared; baselines re-execute frozen code and reproduce saved cells exactly (Appendix E.1).

Table 2: Registration status. *Pre-declared*: protocol frozen (hashed for all post-original cohorts) before its designated seeds ran. *Exploratory*: not declared before outcomes were computed.

	Original ($n = 32$, discovery)		Replication ($n = 252$, primary)		
Paired planning-loss contrast	Mean	95% interval	Mean	95% interval	Verdict at δ
E1 naive descendant gain	-0.00266	$[-0.00605, -0.00066]$	-0.00895	$[-0.01962, -0.00248]$	detected
E1 procedure main effect	-0.00101	$[-0.00271, -0.00002]$	-0.00406	$[-0.00936, -0.00087]$	detected
E1 structure main effect	-0.00165	$[-0.00337, -0.00060]$	-0.00489	$[-0.01029, -0.00150]$	detected
E1 inherited − fixed meta (empty support)	0.00199	$[0.00021, 0.00527]$	0.00032	$[0.00021, 0.00043]$	det. + equiv.
E2 owned × replay interaction	0.00257	$[0.00003, 0.00622]$	0.00987	$[0.00036, 0.02570]$	detected
E3 continued − frozen (package)	-0.01414^*	$[-0.02527, -0.00462]$	-0.01010	$[-0.01501, -0.00579]$	relevant
E3 crossing: code effect	-0.01414	$[-0.02526, -0.00461]$	-0.01029	$[-0.01510, -0.00615]$	relevant
E3 crossing: history effect	-1.84×10^{-6}	$[-5.53 \times 10^{-6}, 0]$	0.00019	$[-0.00002, 0.00060]$	equivalent
E3 continued − evidence-matched baseline	0.00136	$[-0.00026, 0.00408]$	0.00142	$[-0.00059, 0.00370]$	equivalent
E3 continued − validation-selected	0.00411	$[-0.00009, 0.01001]$	0.00262	$[0.00026, 0.00528]$	det. + equiv.
E3 continued − per-task oracle	0.00655	$[0.00186, 0.01257]$	0.00511	$[0.00295, 0.00765]$	detected
E3 continued − fixed meta	-0.00988	$[-0.01961, -0.00208]$	-0.00129	$[-0.00274, 0.00002]$	equivalent
E3 retention: continued − frozen	0.00073	$[-7.38 \times 10^{-7}, 0.00218]$	0.00026	$[0.00004, 0.00058]$	det. + equiv.
Follow-up: continued − frozen	-0.01080^*	$[-0.02201, -0.00264]$	-0.00697	$[-0.01024, -0.00397]$	detected
Follow-up: continued − fixed meta	-0.00013	$[-0.00192, 0.00199]$	-0.00425	$[-0.00704, -0.00187]$	detected
Follow-up: continued − gen.-locked	-0.00086	$[-0.00491, 0.00232]$	0.00031	$[-0.00217, 0.00251]$	equivalent
Follow-up retention: cont. − frozen	0.00007	$[7.52 \times 10^{-7}, 0.00020]$	0.00047	$[0.00007, 0.00104]$	det. + equiv.

Table 3: All primary and extension contrasts under the shared-history contract. Original cohorts: seeds 8101–8132 (E1–E3) and 9401–9432 (follow-up); replications: 8201–8452 and 9501–9752, never pooled. Negative values favor the first-named arm; paired percentile bootstrap over families (10,000 resamples). Verdicts apply $\delta = 0.005$ to the replication: *relevant* = detected with 90% interval outside $\pm\delta$; *detected* = 95% interval excludes zero, relevance not established; *det. + equiv.* = detected but 90% interval inside $\pm\delta$; *equivalent* = not detected and 90% interval inside $\pm\delta$. *Only original-cohort primary that survives Holm adjustment within its cohort. Crossing and baseline rows are post-hoc re-executions of the frozen code.

is detected. Effects are heavy-tailed (median continued-minus-frozen difference -0.00059, mean -0.01010), and cohort size was set by rotation balance, not a power calculation (Appendix E.6).

5.2 Attribution under the shared-history contract

E1: inherited structure, not inherited procedure. *Naive reading.* The inherited model and procedure together improve on the base model and base procedure by -0.00895 ($[-0.01962, -0.00248]$). *Controlled reading.* Crossing base and inherited symbolic support with base and inherited procedures under the same target evidence and fit cap apportions this gain (Equation 1); coefficients are always refitted, so the model factor carries symbolic support only. The procedure main effect is -0.00406 ($[-0.00936, -0.00087]$; 45% of the naive gain) and inherited structure contributes -0.00489 ($[-0.01029, -0.00150]$). The procedure effect survives removal of the three most influential families at about a quarter of its size (-0.00100). On empty support, inherited repair is slightly worse than the source-trained fixed meta-procedure (0.00032, $[0.00021, 0.00043]$; Table 3). The original cohort gives a procedure share of 38%, carried by one family.

E2: owned versus replayed evidence. When each branch chooses its own three rounds of actions from a shared start and owns the outcomes, the inherited procedure's benefit is smaller than under donor replay: the owned-minus-replay interaction is 0.00987 ($[0.00036, 0.02570]$), so replay over-

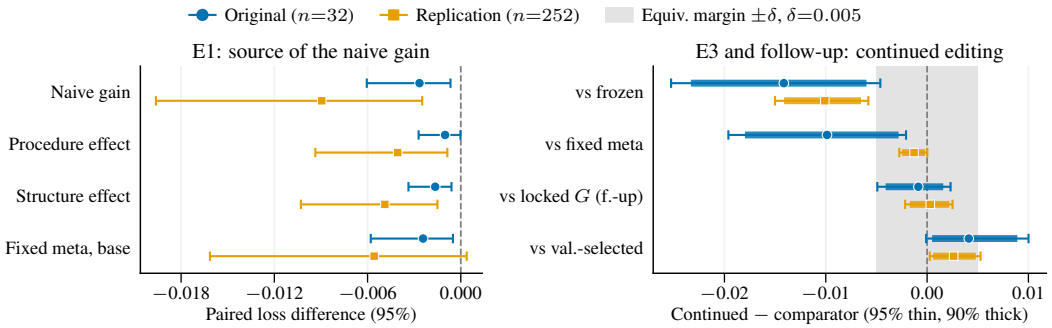

Figure 2: Attribution controls on both cohorts. Left: E1 decomposes the naive descendant gain into procedure and structure main effects, with a fixed meta-procedure applied to the base model for reference. Right: E3 and follow-up contrasts (validation-selected: code×history crossing) with 95% (thin) and 90% (thick) paired bootstrap intervals; the shaded band is the declared equivalence margin ± 0.005. Negative values favor the continued-editing or inherited arm.

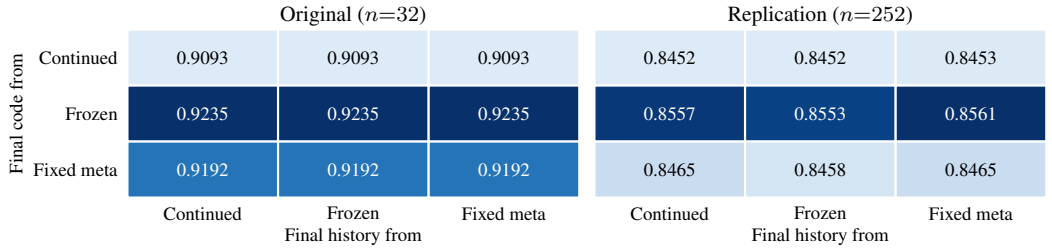

Figure 3: Crossing final code with final history in the original ($n = 32$) and replication ($n = 252$) cohorts. Each cell is the mean transfer loss when one arm's final program repairs with one arm's final history on common transfer evidence. Rows (code) differ; columns (history) do not. Darker cells have higher loss; the colour scale is set per panel.

states the inherited procedure's relative benefit. The effect is concentrated (one family carries 65% of the sum; it remains detected without that family), and the fixed meta-procedure's own and replay outcomes coincide exactly. Because acquisition is not trained, E2 shows that the estimand matters, not that useful experiment selection was learned.

E3: continued editing versus freezing. *Naive reading.* From the same inherited checkpoint, continued editing beats a frozen procedure on held-out triple compositions: -0.01010 ($[-0.01501, -0.00579]$), practically relevant, and again in the follow-up replication (-0.00697). All arms share a cap of 48 fits per continuation stage, but this endpoint evaluates each final program with its branch-specific history.

Controlled reading: the gain is code, and it coincides with leaving single-term repair. Restoring each branch's final code with each branch's final history on common transfer evidence separates the two. The code effect is -0.01029 ($[-0.01510, -0.00615]$), close to the package effect -0.01010; the history effect is 0.00019 ($[-0.00002, 0.00060]$), exactly zero in 247 of 252 families, and practically equivalent (Figure 3; original cohort: 31 of 32 tied). The code that matters is small. The promoted checkpoint uses history-adjusted diagnosis with *single*-term repair in 100% of families, whereas continued editing, on pair-mechanism continuation tasks, ends mostly in paired repair. Continued editing therefore helps by leaving the single-term behavior that the checkpoint adopted, not by accumulating procedural skill. Section 5.4 asks where that behavior comes from.

Is recursion necessary? Non-recursive baselines. We compare the lineage with post-hoc non-recursive search over exactly the six behaviors it can reach. The closest match uses the same evidence: one behavior per family is chosen by its mean validation error on the lineage's own eight

Rung	Effect	Mean	95% interval	90% interval	Verdict
Generator	planted: inert generator	-0.00180	$[-0.00326, -0.00061]$	$[-0.00300, -0.00077]$	det. + equiv.
History	planted: history emptied	-0.00369	$[-0.00621, -0.00132]$	$[-0.00573, -0.00170]$	detected
Code×history	natural: history effect	-0.00172	$[-0.00349, -0.00026]$	$[-0.00318, -0.00046]$	det. + equiv.
Generator	natural: cont. − locked	-0.00260	$[-0.00622, 0.00044]$	$[-0.00546, -2.50 \times 10^{-6}]$	not detected

Table 4: Positive control under test-time self-editing ($n = 252$; pre-declared). Negative values favor the arm with the working generator or history. Under frozen evaluation the same lock and history contrasts are practically equivalent to zero.

continuation tasks, spending 144 fits against the lineage's 192, and then frozen. Continued minus this baseline is 0.00142 ($[-0.00059, 0.00370]$; 90% $[-0.00029, 0.00329]$): practically equivalent, as in the original cohort (0.00136, $[-0.00026, 0.00408]$). Selecting per task by validation error, which spends up to 18 rather than 3 fits per task but no continuation stages, is detected but practically equivalent: 0.00262 ($[0.00026, 0.00528]$; 90% $[0.00060, 0.00480]$). Continued editing never beats the per-task oracle. Against the history-aware fixed meta-procedure, continued editing is practically equivalent in the main replication (-0.00129; 90% $[-0.00248, -0.00019]$) and detected, at about 0.5% of loss, in the follow-up replication (-0.00425, $[-0.00704, -0.00187]$).

Generator plasticity: practically equivalent to locking it. *Naive reading.* The lineage edits its own successor generator, and later generators execute and change the proposals they emit (Appendix C). *Controlled reading.* A fourth follow-up branch keeps the inherited generator fixed while diagnosis, repair and acquisition may still change; the lock applies before candidate truncation and to lookahead descendants, with evidence and fit caps matched. In the replication, continued minus generator-locked is 0.00031 ($[-0.00217, 0.00251]$), and the 90% interval $[-0.00171, 0.00219]$ lies inside ± 0.005, and inside $\pm\delta/2$ as well: generator editing is practically equivalent to locking the generator (original cohort -0.00086, $[-0.00491, 0.00232]$). This is consistent with the six-behavior reduction: under frozen evaluation the generator can matter only through which diagnosis/repair pair a lineage reaches. Retention on source singletons changes detectably but negligibly (Table 3).

5.3 Positive control: can the ladder return a positive verdict?

Under frozen evaluation only diagnosis and repair execute, so the lock and history nulls could be structural. Under test-time self-editing, each final program runs two rounds of its frozen editing step on the transfer task's own train/validation split before the protected repair; every arm spends the same 240 edit fits per family, and nothing else changes. In 252 fresh families two cells plant known effects: a generator restricted to edits that never execute (its cell equals the frozen-regime continued arm exactly), and continued code with its history emptied. With the same contrasts and inference, the ladder detects both planted effects and a natural history effect that is exactly zero in 240 of 252 families under frozen evaluation (Table 4); the natural generator lock is not detected. The planted generator effect is real, yet its 90% interval lies inside $\pm\delta$: our equivalence verdicts bound magnitude and do not assert absence. The 16 pilot seeds were inspected before the predictions were written (Appendix E.3).

5.4 What produces the continued-editing gain

A pre-declared test falsifies a curriculum explanation. A natural explanation attributes the continued-over-frozen gain to curriculum-induced specialization: every source stage exposes one mechanism. A pre-declared test ($n = 252$) that adds pair tasks to the source curriculum falsified this: frozen checkpoints lock to single-term repair under both curricula (252/252), and the gap is unchanged (DiD -0.00029, $[-0.00114, 0.00036]$). The independent-lineage experiment supplies the mechanism (Appendix E.2).

The gain over freezing hinges on the history contract. In every cohort above, the inherited and fixed-meta source lineages write their candidate records into one history. A pre-declared test crossed this contract: in 126 fresh families, 4 source lineages per family differ only in source-task seeds, and under *separate* histories the fixed-meta source lineage keeps its own evidence stream and history. Lineage outcomes are highly redundant (continued transfer is identical across lineages in 64 of 126

Contrast ($n = 126$ families $\times$ 4 lineages)	Shared source history			Separate source histories		
	Mean	95% interval	Verdict	Mean	95% interval	Verdict
Continued − frozen	-0.00902	$[-0.01292, -0.00543]$	relevant	-0.00121	$[-0.00316, 0.00121]$	equivalent
Continued − fixed meta	-0.00302	$[-0.00550, -0.00079]$	detected	-0.00098	$[-0.00288, 0.00141]$	equivalent
Continued − validation-selected	0.00442	$[0.00157, 0.00790]$	detected	0.00360	$[0.00079, 0.00711]$	detected
Checkpoints with single-term repair	504 of 504			75 of 504		

Table 5: Independent source lineages (pre-declared; families are the unit of inference). Only the history contract between the inherited and fixed-meta source lineages differs between the two halves; neither half is an unbiased reference. Verdicts at $\delta = 0.005$ as in Section 5.1.

shared and 100 of 126 separate families), so lineages are averaged within family before the bootstrap; Holm covers the six primaries. Under shared history the replication pattern returns (Table 5). Under separate histories only 75 of 504 checkpoints use single-term repair (365 are residual/paired), the frozen checkpoint improves by -0.00831 ($[-0.01178, -0.00508]$), and continued minus frozen is no longer detected (Holm $p = 0.56$); the factor difference is 0.00782 ($[0.00404, 0.01258]$). A trace of the source search identifies the mechanism (Appendix E.4): the base generator starts its proposal window at the history length, so the extra records shift which edits the inherited lineage is offered, and content-free placeholder records of the same count reproduce all 504 shared checkpoints. Neither regime is the natural one, since both obey the same length rule. The pair shows that the continued-over-frozen gain is fragile, hinging on an incidental design feature of exactly the kind Section 3 requires comparisons to state, whereas non-recursive per-task validation selection is detectably better than continued editing in both. At equal lifetime fit budget, continued editing is detectably better than validation selection only after about 80 tasks under shared history; under separate histories the frozen checkpoint without continuation matches it (0.8726 versus 0.8714; Appendix E.5).

6 Discussion, Recommendations and Limitations

A descendant curve in this testbed tells a clear story of recursive self-improvement, and the ladder assigns most of it to other causes: inherited model structure, replayed evidence, and a final-code change whose value over freezing hinges on an incidental checkpoint detail and is matched by non-recursive selection. Executable recursion and useful recursion are different claims; only matched interventions separate them.

Recommendations. (1) Cross inherited model/state with inherited procedure under common evidence and report the procedure main effect, not the diagonal. (2) Restore each final program with each history before crediting code, and rerun the key contrast under a second checkpoint contract (what lineages share, what a history contains). (3) Lock only the recursive component, state an equivalence margin in advance, and plant a known effect to show the test can detect one. (4) Enumerate the behaviors that execute at evaluation and compare against non-recursive choices among them on the same evidence. (5) Own the evidence, keep the final evaluator outside the search loop, and declare independent instances as the unit of inference before execution.

Limitations. The testbed is a toy by design: a one-dimensional synthetic system whose simulator and model share a realizable symbolic dictionary, and a finite grammar of 54 programs; transfer recombines known primitives. Frozen-regime nulls are partly structural (six behaviors); the positive control detects such effects when they execute, but detectable effects can be smaller than δ, so our equivalence statements bound magnitude and do not assert absence. When the simulator adds a velocity term the dictionary cannot represent ($n = 252$; Appendix D.6), the attribution pattern persists (procedure share 38%; continued minus frozen -0.00757, $[-0.01264, -0.00237]$), but continued editing is no longer detectably better than the fixed meta-procedure (0.00128, $[-0.00261, 0.00577]$). We compare against baselines within the grammar, not against external self-improving agents. The replications and later tests were designed after the original results were known, although each was declared and hashed before execution; the code$\times$history crossing and behavior baselines are undeclared post-hoc analyses (Table 2). The findings are claims about this lineage; each names a control a stronger system must pass (extensions: Appendix H).

AI USE STATEMENT

Large language model agents assisted with conceptual development, literature analysis, hypothesis and experimental design, implementation, analysis and figure scripting, interpretation, LaTeX drafting, wording and consistency revision, and citation formatting. Experimental observations and numerical results were obtained by executing the disclosed synthetic simulators and computational procedures; they were not supplied directly by an LLM. Reproducibility materials include frozen protocol and code manifests, raw records, and analysis scripts. All AI-assisted text and code were reviewed by the authors; the authors take responsibility for the final text, code, results, and disclosures.

ETHICS STATEMENT

The study concerns controlled computational environments. Its simulated dynamics and decision costs do not establish safe operation of physical equipment, prospective scientific discovery, biological efficacy or clinical benefit. The separation between editable procedures and a fixed evaluation kernel is an experimental access boundary, not a general safety guarantee.

REPRODUCIBILITY STATEMENT

The supplement supplies source code; the frozen original protocols; the replication, misspecification, curriculum-intervention, positive-control and independent-lineage protocols, each with its SHA-256 freeze manifest and timestamp (each declared before its designated seeds were executed, with the positive-control pilot disclosed); the post-hoc code×history crossing and evidence-matched baseline scripts, which re-execute the frozen code, assert exact reproduction of every saved diagonal cell and check the six-behavior reduction on the first four families; raw trial and lineage records; analysis scripts; plotted data; and editable vector figures. A world family runs in about 1.5 s on a laptop CPU, so every cohort can be regenerated in minutes. Uncertainty is conditional on the specified synthetic mechanism distribution. The appendix describes data roles, model and procedure grammars, planning, branch ownership, costs and controls.

REFERENCES

Don Joven Agravante, Daiki Kimura, Michiaki Tatsubori, Asim Munawar, and Alexander Gray. Learning neuro-symbolic world models with conversational proprioception. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 648–656, 2023. doi: 10.18653/v1/2023.acl-short.57.

Marcin Andrychowicz, Misha Denil, Sergio Gómez, Matthew W. Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/fb87582825f9d28a8d42c5e5e5e8b23d-Abstract.html.

Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic Tensor Networks. *Artificial Intelligence*, 303:103649, 2022. doi: 10.1016/j.artint.2021.103649. URL https://arxiv.org/abs/2012.13635.

Jonathan C. Balloch, Zhiyu Lin, Robert Wright, Xiangyu Peng, Mustafa Hussain, Aarun Srinivas, Julia M. Kim, and Mark O. Riedl. Neuro-symbolic world models for adapting to open world novelty, 2023. URL https://arxiv.org/html/2301.06294v1.

Xiaoyi Bao, Yuanzhen Xie, Yunzhi Tan, Jinghang Gu, Zhongqing Wang, Chu-Ren Huang, Bo Hu, and Zang Li. SkillMentor: LLM agent self-evolution via learning blind-spot diagnosis, 2026. URL https://arxiv.org/html/2607.27360v1.

Kexin Gu Baugh, Luke Dickens, and Alessandra Russo. Neural DNF-MT: A neuro-symbolic approach for learning interpretable and editable policies. In *International Conference on Autonomous Agents and Multiagent Systems*, 2025. URL https://arxiv.org/abs/2501.03888v4.

Lawrence Birnbaum, Gregg Collins, Michael Freed, and Bruce Krulwich. Model-based diagnosis of planning failures. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pp. 318–323, 1990. URL https://cdn.aaai.org/AAAI/1990/AAAI90-049.pdf.

Avrim Blum and Moritz Hardt. The ladder: A reliable leaderboard for machine learning competitions. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1006–1014, 2015. URL https://proceedings.mlr.press/v37/blum15.html.

Akash Bonagiri, Devang Borkar, Gerard Janno Anderias, Setareh Rafatirad, and Houman Homayoun. CausalFlow: Causal attribution and counterfactual repair for LLM agent failures, 2026. URL https://arxiv.org/abs/2605.25338.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113:3932–3937, 2016. doi: 10.1073/pnas.1517384113. URL https://arxiv.org/abs/1509.03580v1.

Gavin C. Cawley and Nicola L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11(70):2079–2107, 2010. URL https://jmlr.org/papers/v11/cawley10a.html.

Cristina Cornelio and Mohammed Diab. Recover: A neuro-symbolic framework for failure detection and recovery, 2024. URL https://arxiv.org/abs/2404.00756v1.

Michael T. Cox and Ashwin Ram. Introspective multistrategy learning: On the construction of learning strategies. *Artificial Intelligence*, 112:1–55, 1999. URL https://sites.cc.gatech.edu/fac/ashwin/papers/er-99-01.pdf.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30, 2006. URL https://jmlr.org/papers/v7/demsar06a.html.

Thomas G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, 1998. doi: 10.1162/089976698300017197.

Lin Du, Jie Zhou, Yuxuan Cai, Kai Chen, Qin Chen, Xin Li, Bo Zhang, Wei Li, and Liang He. AutoMem: A text-gradient recursive self-improvement framework for automated memory architectures search, 2026. URL https://arxiv.org/html/2608.14621v1.

Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. In *Proceedings of the 28th International Conference on Machine Learning*, 2011. URL https://icml.cc/2011/papers/554_icmlpaper.pdf.

Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Roth. The reusable holdout: Preserving validity in adaptive data analysis. *Science*, 2015. doi: 10.1126/science.aaa9375.

Kevin Ellis, Catherine Wong, Maxwell Nye, Mathias Sable-Meyer, Luc Cary, Lucas Morales, Luke Hewitt, Armando Solar-Lezama, and Joshua B. Tenenbaum. DreamCoder: Growing generalizable, interpretable knowledge with wake-sleep Bayesian program learning, 2020. URL https://arxiv.org/abs/2006.08381v1.

Tianyu Fan and Chao Huang. HELIX: Model-harness co-evolution for recursive self-improvement, 2026. URL https://arxiv.org/abs/2608.13951v1.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1126–1135, 2017. URL https://proceedings.mlr.press/v70/finn17a.html.

Adam Foster, Desi R Ivanova, Ilyas Malik, and Tom Rainforth. Deep Adaptive Design: Amortizing sequential Bayesian experimental design. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3384–3395, 2021. URL https://proceedings.mlr.press/v139/foster21a.html.

Susan Fox. A new model of reflective introspective learning. In *Proceedings of the Eleventh International FLAIRS Conference*, pp. 259–263, 1998. URL https://cdn.aaai.org/FLAIRS/1998/FLAIRS98-050.pdf.

Jiarun Fu, Lizhong Ding, Sida Chen, Honglei Xin, Chunhui Zhang, Pengqi Li, Qiuning Wei, Ye Yuan, and Guoren Wang. CHILL-Harness: Counterfactual harness learning for efficient reasoning in long-horizon agents, 2026. URL https://arxiv.org/abs/2607.25825v1.

David Ha and Jürgen Schmidhuber. World Models, 2018. URL https://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565, 2019. URL https://proceedings.mlr.press/v97/hafner19a.html.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.02193v4.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104v2.

Safayat Bin Hakim, Keyan Guo, Wenkai Tan, Alvaro Velasquez, Shouhuai Xu, and Houbing Herbert Song. ANNEAL: Adapting LLM agents via governed symbolic patch learning, 2026. URL https://arxiv.org/abs/2605.16309v1.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.16828v2.

Wen-Chao Hu, Wang-Zhou Dai, Yuan Jiang, and Zhi-Hua Zhou. Efficient rectification of neuro-symbolic reasoning inconsistencies by abductive reflection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17333–17341, 2025. doi: 10.1609/aaai.v39i16.33905.

Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70:489–501, 2006. doi: 10.1016/j.neucom.2005.12.126. URL https://extreme-learning-machines.org/pdf/ELM-NC-2006.pdf.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 652–661, 2016. URL https://proceedings.mlr.press/v48/jiang16.html.

Minsoo Kim and Seung won Hwang. CoEx: Co-evolving world-model and exploration, 2025. URL https://aclanthology.org/2025.findings-emnlp.1179.pdf.

Claire Le Goues, ThanhVu Nguyen, Stephanie Forrest, and Westley Weimer. GenProg: A generic method for automatic software repair. *IEEE Transactions on Software Engineering*, 38(1):54–72, 2012. doi: 10.1109/TSE.2011.104. URL https://clairelegoues.com/assets/papers/genprog12tse.pdf.

Yuqing Li, Zeguan Wu, Yu Gan, and Junyu Liu. Self-modifying Lean proof agents with verifier-grounded benchmark coevolution, 2026. URL https://arxiv.org/html/2607.17352v1.

Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-SGD: Learning to learn quickly for few-shot learning, 2017. URL https://arxiv.org/abs/1707.09835.

Yichao Liang, Nishanth Kumar, Hao Tang, Adrian Weller, Joshua B. Tenenbaum, Tom Silver, João F. Henriques, and Kevin Ellis. VisualPredicator: Learning abstract world models with neuro-symbolic predicates for robot planning, 2025. URL https://arxiv.org/html/2410.23156v2.

Changzhi Liu, Yilun Liu, Sikuan Yan, Volker Tresp, and Yunpu Ma. Mendel Gödel Machine: Recursive self-improving coding agents via comparative evolution, 2026. URL https://arxiv.org/abs/2608.07645.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2019.

Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. DeepProbLog: Neural probabilistic logic programming, 2018. URL https://arxiv.org/abs/1805.10872. Accepted at NeurIPS 2018.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The Neuro-Symbolic Concept Learner: Interpreting scenes, words, and sentences from natural supervision, 2019. URL https://arxiv.org/abs/1904.12584.

Sergey Mechtaev, Jooyong Yi, and Abhik Roychoudhury. Angelix: Scalable multiline program patch synthesis via symbolic analysis. In *Proceedings of the 38th International Conference on Software Engineering*, pp. 691–701, 2016. doi: 10.1145/2884781.2884807. URL https://discovery.ucl.ac.uk/id/eprint/10088929/.

Luke Metz, C. Daniel Freeman, Niru Maheswaranathan, and Jascha Sohl-Dickstein. Training learned optimizers with randomly initialized learned optimizers, 2021. URL https://arxiv.org/abs/2101.07367.

Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Machine Learning*, 52: 239–281, 2003. doi: 10.1023/A:1024068626366.

Hoang Duong Thien Nguyen, Dawei Qi, Abhik Roychoudhury, and Satish Chandra. SemFix: Program repair via semantic analysis. In *Proceedings of the 35th International Conference on Software Engineering*, 2013. URL https://2013.icse-conferences.org/content/semfix-program-repair-semantic-analysis.html.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms, 2018. URL https://arxiv.org/abs/1803.02999.

Jun Nie, Yonggang Zhang, Jun Song, Qianshu Cai, Dahai Yu, Yike Guo, Xinmei Tian, and Bo Han. TTHE: Test-time harness evolution, 2026. URL https://arxiv.org/html/2607.08124v1.

Top Piriyakulkij, Yichao Liang, Hao Tang, Adrian Weller, Marta Kryven, and Kevin Ellis. PoE-World: Compositional world modeling with products of programmatic experts. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0896.

Esteban Real, Chen Liang, David So, and Quoc Le. AutoML-Zero: Evolving machine learning algorithms from scratch. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8007–8019, 2020. URL https://proceedings.mlr.press/v119/real20a.html.

Jürgen Schmidhuber. Gödel machines: Fully self-referential optimal universal self-improvers. In Ben Goertzel and Cassio Pennachin (eds.), *Artificial General Intelligence*, pp. 199–226. Springer, 2007. doi: 10.1007/978-3-540-68677-4_7.

Atharva Sehgal, Arya Grayeli, Jennifer J. Sun, and Swarat Chaudhuri. Neurosymbolic grounding for compositional world models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.12690v2.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0377.

James E. Smith and Robert L. Winkler. The optimizer's curse: Skepticism and postdecision surprise in decision analysis. *Management Science*, 52(3):311–322, 2006. doi: 10.1287/mnsc.1050.0451.

Adith Swaminathan and Thorsten Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 814–823, 2015. URL https://proceedings.mlr.press/v37/swaminathan15.html.

Hao Tang, Darren Key, and Kevin Ellis. WorldCoder, a model-based LLM agent: Building world models by writing code and interacting with the environment, 2024. URL https://arxiv.org/abs/2402.12275.

Sudhir Varma and Richard Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7:91, 2006. doi: 10.1186/1471-2105-7-91.

Wenyi Wang, Piotr Piękos, Li Nanbo, Firas Laakom, Yimeng Chen, Mateusz Ostaszewski, Mingchen Zhuge, and Jürgen Schmidhuber. Huxley-Gödel Machine: Human-level coding agent development by an approximation of the optimal self-improving machine, 2025. URL https://arxiv.org/abs/2510.21614.

Zefeng Wang, Minxi Yan, Jinhe Bi, Sikuan Yan, Volker Tresp, and Yunpu Ma. MetaSkill-Evolve: Recursive self-improvement of LLM agents via two-timescale meta-skill evolution, 2026. URL https://arxiv.org/abs/2607.05297.

Westley Weimer, ThanhVu Nguyen, Claire Le Goues, and Stephanie Forrest. Automatically finding patches using genetic programming. In *Proceedings of the 31st International Conference on Software Engineering*, pp. 364–374, 2009. doi: 10.1109/ICSE.2009.5070536. URL https://clairelegoues.com/assets/papers/weimer09icse.pdf.

Xuening Wu, Shenqin Yin, Yanlan Kang, Xinhang Zhang, Qianya Xu, Zeping Chen, and Wenqiang Zhang. SGM: A statistical Godel machine for risk-controlled recursive self-modification, 2026. URL https://arxiv.org/abs/2510.10232v2.

Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Van den Broeck. A semantic loss function for deep learning with symbolic knowledge. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5502–5511, 2018. URL https://proceedings.mlr.press/v80/xu18h.html.

Qiang Yang, Kangheng Wu, and Yunfei Jiang. Learning action models from plan examples using weighted MAX-SAT. *Artificial Intelligence*, 171(2–3):107–143, 2007. doi: 10.1016/j.artint.2006.11.005.

Eric Zelikman, Eliana Lorch, Lester Mackey, and Adam Tauman Kalai. Self-taught optimizer (STOP): Recursively self-improving code generation. In *Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2310.02304v3.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin Godel Machine: Open-ended evolution of self-improving agents, 2025. URL https://arxiv.org/abs/2505.22954v3.

Jenny Zhang, Bingchen Zhao, Wannan Yang, Jakob Foerster, Jeff Clune, Minqi Jiang, Sam Devlin, and Tatiana Shavrina. Hyperagents, 2026a. URL https://arxiv.org/abs/2603.19461.

Jiayi Zhang, Yongfeng Gu, Jianhao Ruan, Maojia Song, Yiran Peng, Zhiguang Han, Jinyu Xiang, Zhitao Wang, Caiyin Yang, Yixi Ouyang, Bang Liu, Chenglin Wu, and Yuyu Luo. Harnessing agentic evolution, 2026b. URL https://arxiv.org/html/2605.13821v1.

Tong Zheng, Xidong Wu, Zheng Zhang, Zhankui He, Chaoyi Zhang, Benjamin Coleman, Ruoqiao Wei, Di Bai, Haolin Liu, Rui Liu, Xue Wang, Yue Zhuan, Wang-Cheng Kang, Renkai Xiang, Heng Huang, Xinwu Cheng, and Yunsong Guo. Dream-RSI: Recursive self-improvement through evolving worlds, 2026. URL https://arxiv.org/abs/2609.14858.

Tailin Zhou. Hierarchical self-improvement: A framework for task-specific evolvable agent harnesses, 2026. URL https://arxiv.org/abs/2608.08466v1.

A IMPLEMENTATION AND EXPERIMENTAL PROTOCOL

All observations in the three experiments are generated by the supplied simulator. There is no measured scientific dataset, pretrained language-model call, autonomous discovery of primitives, or physical experiment. The implementation is a reference mechanism for the attribution questions in Section 3. Its executable grammar and known dynamics family allow inspection of each candidate and exact evidence replay. Table 6 specifies the editable objects.

A.1 DYNAMICS, OBSERVATIONS AND THE LEARNED COMPONENT

The context features and symbolic terms are

$$\phi(c) = (1, \tanh(1.3c), \tanh(-.8c + .4), \tanh(2.2c - .3)), \tag{4}$$

$$\psi(x, v, u) = (-\tanh(2v), -x^3, \operatorname{sign}(u)\max(|u| - .55, 0) - u, -\max(x, 0)). \tag{5}$$

For each independent seed, the eight base coefficients $(.42, .09, -.05, .03, .16, .055, .30, .16)$ are independently multiplied by uniform draws on $[.75, 1.25]$. A four-bit mask activates symbolic terms. All masks within a family share that family's coefficient vector. Source masks activate each primitive separately; continuation masks activate pairs; transfer masks activate triples. Because coefficients and the complete feature library are shared, the transfer test asks about combinations of familiar mechanisms rather than extrapolation to a new parameter distribution or an unknown functional form.

Evidence contains tuples (x, v, c, u, x', v'). Ordinary training states sample $x \in [-1.8, 1.8]$, $v \in [-.8, .8]$ and $c \in [-1, 1]$ uniformly, with actions drawn from $\{-1, -.35, 0, .35, 1\}$. The initial acquisition evidence instead samples $x \in [-.3, .3]$, $v \in [-.12, .12]$ and actions $\{-.35, .35\}$. Training, validation and prediction-test observations add independent Gaussian noise with standard deviation .005 to next velocity; control evaluation rolls out deterministic dynamics. Next position follows the simulator's unnoised transition. The noise model is deliberately simple; no claim about realistic sensor coupling follows from it.

For a proposed mask, the regression target subtracts the known nominal term $.52v - .10x$. The design matrix contains the four context features multiplied by the action, together with active symbolic terms. Ridge regularization is 10^{-3}. Every call refits all active coefficients. Thus an inherited model entering a repair carries its symbolic support, but its numerical initialization does not survive that call. The system learns a linear output layer over fixed nonlinear features. It has neither a trained perceptual encoder nor a learned hidden-state representation.

Component	Editable or learned	Fixed boundary
World model	Symbolic support; output coefficients.	Nominal transition and full feature dictionary.
Diagnosis	One of three executable ranking functions.	Residual/correlation interface and four candidate terms.
Repair	Single or paired support edits.	Ridge solver, evidence partition and fit accounting.
Acquisition	One of three action selectors.	Action set, reachable simulator and ensemble construction.
Successor generation	One of three proposal functions.	Allowed subtree grammar and candidate budget.
Promotion	Candidate identity changes after improvement.	Validation objective and strict acceptance rule.
Planning/evaluation	Model supplied to planning changes.	Horizon, cost, protected evaluation seeds.

Table 6: The implementation's edit boundary. An editable choice inside a fixed grammar is real procedural state, but does not make the grammar or evaluator editable.

A.2 DIAGNOSIS, REPAIR AND PROCEDURAL CREDIT

Algorithms A1 and A2 specify model repair and successor evaluation. Diagnosis receives residual-feature correlations and accessible history. The residual option orders absolute correlations; the his-

tory option adds a scaled average of earlier gains associated with changed mechanism bits. Cyclic diagnosis ignores these scores. Correlation is a candidate-ordering heuristic, not a causal explanation. Repair toggles one bit, or proposes adjacent pairs in the diagnostic order interleaved with single toggles. It evaluates the incumbent and at most two unique alternative supports. A lower validation error accepts a model candidate; rejected fits still consume resources.

The 54 executable procedures are the product of three diagnoses, two repair enumerators, three acquisition selectors and three successor generators. Legal successors differ in one or two function subtrees. Cyclic generation rotates a legal enumeration using history length. Credit generation ranks edit signatures by observed improvement with an exploration bonus. Composition generation prioritizes two-subtree edits when recent residual validation error exceeds its fixed threshold. Each emitted procedure has a source hash, and the log retains the parent, candidate, generated descendant, evidence hashes and scores.

Procedure evaluation uses two tasks, each with 36 fitting and 24 validation transitions. The parent candidate repairs a model on the first task. Its generator then produces one child, and the child repairs a model on the second task. The mean of these validation losses is the development objective. The fixed-meta control uses its own fixed proposal function in both the outer search and descendant lookahead. It pools credit and deduplicates candidate keys by identical diagnosis/repair choices. Acquisition and inherited-generator fields are omitted from that key, although the complete program can still affect deterministic proposal tie ordering. This is a chosen search heuristic, not established whole-program equivalence. Four candidates plus the incumbent cost 30 ridge solves, leaving 18 of the 48 continuation credits for model repair. The frozen branch can spend all 48 on repair. Equal fit credits do not imply equal useful information or equal wall time; a converged deterministic repairer can repeat the same search.

Algorithm A1: bounded model repair

1. Refit the incumbent support on fitting evidence; score on separate validation evidence.
2. Compute residual-feature correlations; execute the procedure's diagnosis function with the permitted history.
3. Execute its repair enumerator. Skip duplicate supports and stop when the fit budget is exhausted.
4. Fit and validate each proposed support, retaining a strictly better incumbent.
5. Return the selected executable model and a trace of every candidate, score, changed bit and evidence identity.

Algorithm A2: successor evaluation and promotion

1. Execute the active generator to propose four legal procedures from the common history.
2. For each candidate and the incumbent, repair the first task with three fits.
3. Execute that candidate's generator to emit a descendant; repair the second task with three fits. For the fixed-meta control, execute its fixed generator here as well.
4. Average the two validation losses. Promote the best candidate only if its reduction exceeds 10^{-8}.
5. Append all candidate records, including rejected edits, to the accessible history. Preserve code and cost identities before continuing.

A.3 Planning outcomes and experiment partitions

The planner exhaustively scores all 5^3 three-step action sequences. Predicted per-step cost is $(x - g)^2 + .10v^2 + .025u^2$; only the first action is executed before replanning. Evaluation averages 10 executed steps from each of eight independently sampled initial states, with alternating goals $g = \pm 1.2$. A separate set of 80 transitions measures next-velocity mean squared error. Planning uses

the repaired model, while evaluation transitions use the held-out simulator. Test results are saved for analysis and are not read by proposal or promotion functions. Raw history also retains audit labels such as seed, stage and mechanism mask. The supplied functions access only the documented score and credit fields; perturbing unused metadata checks this property. There is no adversarial access-control sandbox.

Source learning performs four stages, one per singleton mechanism. Inherited and fixed-meta source lineages see the same task evidence and a shared history containing both lineages' candidate records and their raw task evidence. This sharing equalizes their accessible diagnostic information; it also means the inherited source lineage is not an isolated autonomous learner. After source training, Experiment 1 crosses empty versus inherited support with frozen procedure variants on a pair-mechanism task. The base and inherited cells define the factorial effects; fixed history-aware and source-trained fixed-meta procedures provide additional comparisons.

Experiment 2 initializes every branch with the same 12 benign transitions and 24 validation transitions. Each adaptive round acquires 12 successive reachable transitions, repairs the model and saves a protected evaluation. The initial branch state is $x = 1.2$, $v = 0$, with context $c = \pm.7$. The state evolves according to selected actions; it is not reset to an arbitrary query point. The base branch supplies three donor batches. Every replay branch receives those exact batches in order. Its environment need not be counterfactually reconstructed because the replay estimand is explicitly conditional on the supplied tuples. The base adaptive and base replay outcomes must agree exactly. Action-selection randomness and measurement noise use separate streams.

Experiment 3 branches from the same inherited program and history. Four rotated pair-mechanism tasks supply continuation evidence. Every arm then faces all four triple-mechanism tasks and all four source singletons, beginning each final task with empty model support and fresh common evidence. Final procedures are frozen during these tasks, but their accumulated histories differ because their continuation computations differ. The endpoint therefore measures the inherited procedural package, including accessible history, rather than code alone. The checkpoint retention reference uses the source procedure and source history on the same singleton evidence.

Experiment	Shared at comparison start	Allowed divergence
Crossed repair	Target tuples, validation, history and fit cap.	Input symbolic support and frozen procedure.
Adaptive/replay	Initial model, initial evidence and branch state.	Own actions and observations, or exact donor tuples.
Continued editing	Inherited program, source history and model.	Later programs, repair histories and candidate allocation.
Final transfer	Fresh task evidence and repair cap.	Continued branch's frozen program and inherited history.
Retention	Singleton evidence and final repair cap.	Checkpoint versus later procedural package.

Table 7: Equality contracts and permitted differences. Equal starting evidence does not imply equal future evidence or equal later histories.

Table 7 summarizes the equality contracts. The representation diagnostic fits the complete supplied support and then zeros either its neural output coefficients or symbolic coefficients. These destructive interventions check that both components can affect outcomes. They are not retrained ablations, capacity-matched competitors, or evidence that this hybrid outperforms an unconstrained neural model. The implementation, raw traces and saved evidence are needed to interpret this diagnostic alongside the three primary experiments.

B Detailed outcomes and diagnostic analyses

Table 3 supplies the designated-run contrasts. The following diagnostics describe the same 32 independent families; they do not constitute additional independent experiments.

B.1 Model structure, procedure and prediction quality

The inherited-support cells have a smaller spread between stronger procedures than the empty-support cells. Averaging both contexts prevents choosing the more favorable start. Figure 4 also separates prediction from control: different fitted models can select the same discrete action, so error reductions need not produce proportional planning gains.

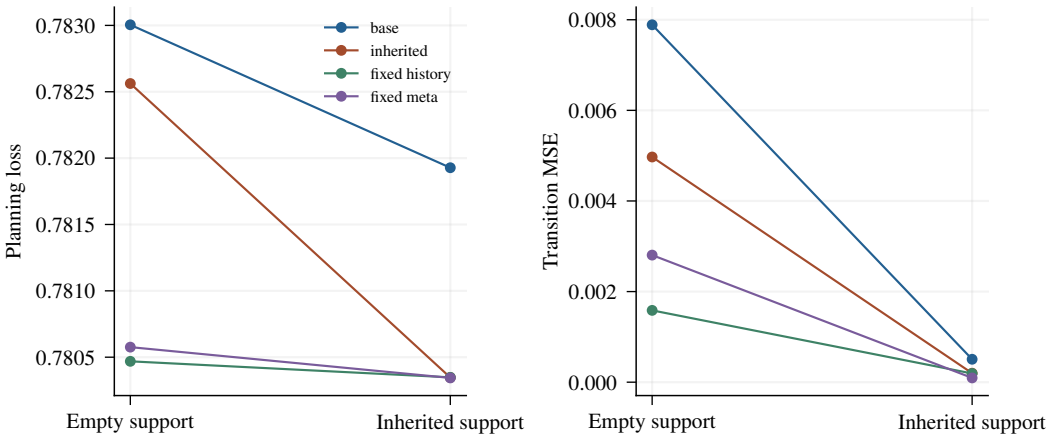

Figure 4: Experiment 1 cell means for all four frozen procedures. The x-axis is the symbolic support entering repair; coefficients are always refitted. Both panels average the same 32 families. Lines connect comparison cells, not training trajectories.

B.2 Sequential evidence and concentration of effects

Figure 5 locates divergence across successive batches. Owned branches encounter their reachable trajectories; replay supplies the base donor's tuples. Rounds are correlated measurements, not independent samples.

The E2 interaction is concentrated in a few families. Its positive mean interval does not imply that replay changes most comparisons. Saved family contrasts expose this concentration without redefining the original endpoint.

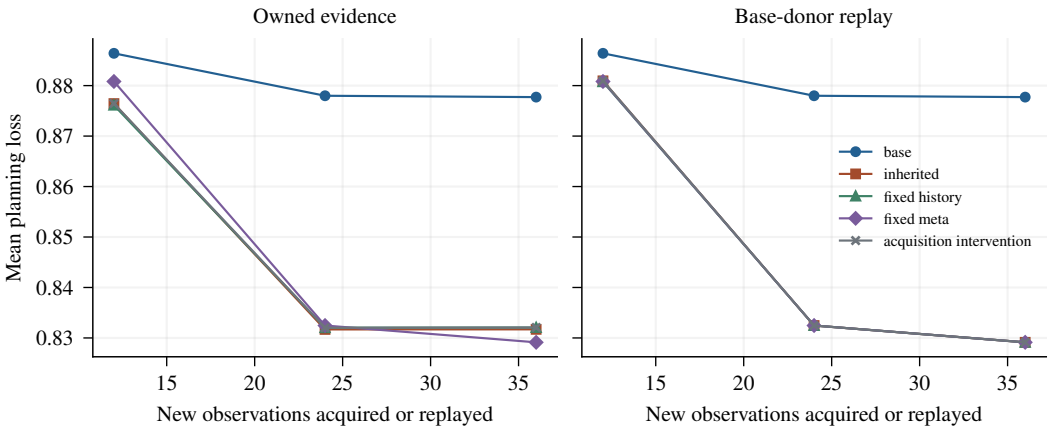

Figure 5: Experiment 2 planning outcomes after each new batch. The horizontal axis excludes the common initial 12 observations. Curves are descriptive family means, with the primary paired inference reserved for the final round. Coincident curves are retained rather than removed as redundant.

B.3 TRANSFER, RETENTION AND THE MEANING OF A HELD-OUT TASK

Transfer allows three model fits on fresh evidence, so it measures adaptation to withheld compositions rather than zero-shot prediction. Figure 6 uses the same contract on source singletons. The continued branch's interval permits a retention cost; improved transfer does not establish monotonic learning.

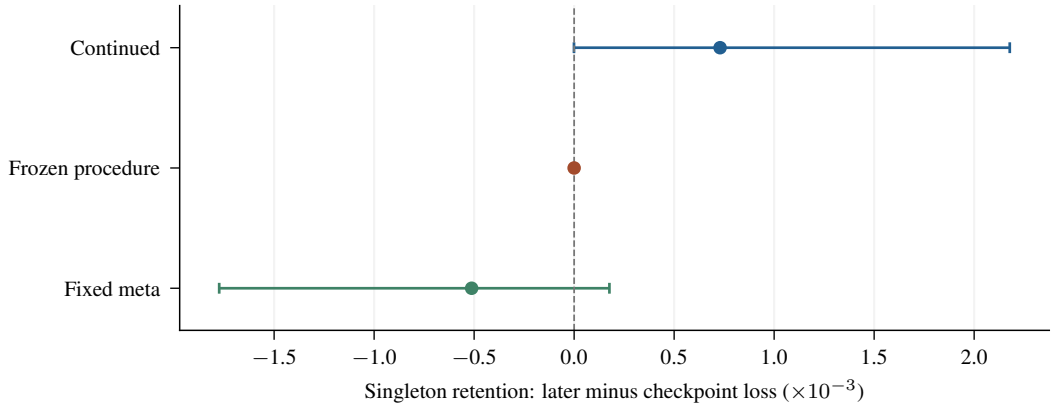

Figure 6: Source-mechanism retention after continuation. Positive differences indicate worse planning than the original checkpoint. The frozen procedure reproduces its checkpoint outcome in these cases; this observed agreement is not an equivalence theorem for all possible histories.

B.4 DO BOTH MODEL COMPONENTS AFFECT BEHAVIOR?

Figure 7 zeros components of a common complete-support fit. Both removals change outcomes; removing the neural action term particularly harms control, as expected from this simulator's actuation law. Without retraining, these are dependence checks, not comparisons with optimized neural-only or symbolic-only learners.

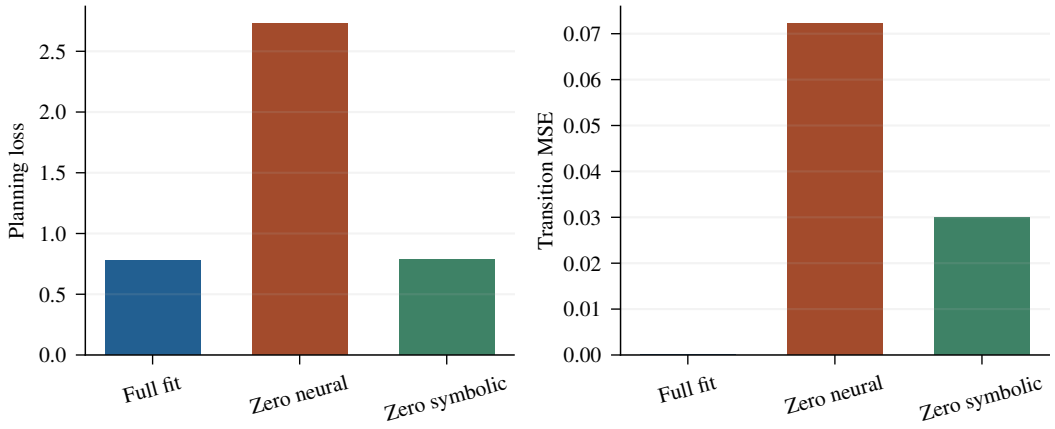

Figure 7: Constructive representation dependence check. Coefficients are zeroed after a common full-support fit. These are destructive interventions without retraining; the bars are mean outcomes, not competitive representation baselines.

B.5 COMPUTE AND LINEAGE EVIDENCE

All continuation arms use 192 ridge solves. Figure 8 shows why this does not imply equal runtime. The ledger separately counts planner-model and true-environment transitions. Source learning

jointly trains two lineages with pooled evidence, so its cost cannot be attributed as one lineage's independent training cost or an established amortized saving.

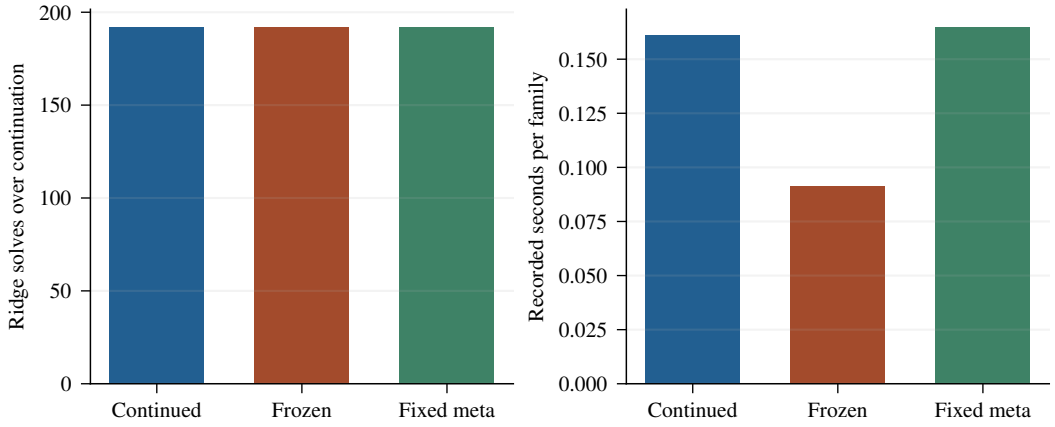

Figure 8: Continuation resource accounting. All arms exhaust the same ridge-solve allowance. Runtime differs despite this equality, and repeated fits can revisit the same candidates. The plot does not claim equal FLOPs, equal useful search, or a lifetime compute advantage.

Figure 9 records accepted generator changes. The lineage table additionally links accepted code to later stage-parent execution, excluding terminal edits from that witness. Generator-only development interventions reconstruct changed descendants. These demonstrate an executed mechanism, not the final utility of a particular edit.

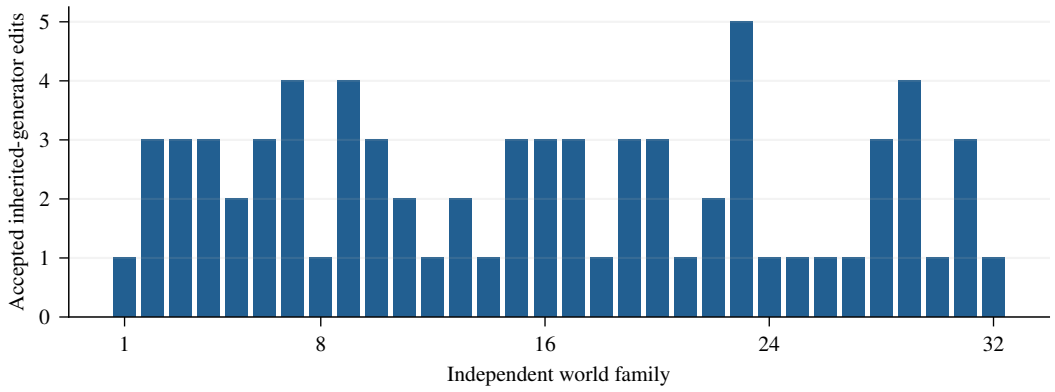

Figure 9: Accepted inherited-generator changes across source and continuation stages. Each bar is one independent family. The accompanying lineage records distinguish acceptance from later execution and retain terminal edits that have no subsequent stage.

B.6 Numerical reconstruction and audit boundaries

An internal audit separately reconstructs regression matrices and validation losses from saved evidence, regenerates the recorded planning endpoints from world coefficients, and recomputes paired summaries with the declared family unit. Corrupted fixtures alter code identities, evidence hashes, promotion scores and replay records to test whether the checker rejects inconsistent evidence. These checks provide concrete error detection; they are not an external scientific review or a proof that every possible implementation fault is absent. The artifact records the exact checks, source hashes and results so that a separate auditor can challenge them.

C A FRESH-SAMPLE TEST OF SUCCESSOR-GENERATOR EDITING

The initial Experiment 3 changes diagnosis, repair and generation jointly. A follow-up motivated by those outcomes therefore locks only the generator. This exploratory extension was separately frozen before fresh-family execution. It leaves the initial source unchanged and does not pool the two samples.

C.1 INTERVENTION AND COMPARISON CONTRACT

Fresh seeds 9401–9432 produce 32 new coefficient families under the same simulator distribution. The source-learning procedure is unchanged. Each family supplies its inherited checkpoint to four branches: continued editing, a fully frozen procedure, the fixed history-aware meta-procedure, and a generator-locked procedure. The last branch may edit diagnosis, model-repair enumeration and acquisition, but retains its checkpoint generator (Table 8).

The lock restricts the legal edit set before the active generator chooses its four candidates. It is applied again when each candidate emits its lookahead descendant. Thus a descendant cannot evade the lock by changing the generator at an intermediate stage. The retained generator still executes and responds to history; the intervention freezes its code, not its outputs. Every branch receives the same raw task evidence and the same 48-fit continuation cap. Final transfer and retention still begin each task with common fresh evidence, empty model support and three model fits.

Branch	Procedural permissions	Interpretation of continued-minus-branch
Frozen procedure	No procedure subtrees change; model repair continues.	Benefit of continued procedural search over the inherited package.
Fixed meta	Immutable external proposer; active repair components can change.	Comparison with one strong fixed adaptive search rule.
Generator locked	Diagnosis, repair and acquisition can change; inherited generator code cannot.	Increment from permitting generator edits within this grammar and search rule.
Continued	All four procedure subtrees can change.	Reference treatment; not unrestricted source synthesis.

Table 8: The follow-up separates a frozen procedure from a frozen generator. A fixed generator remains history-responsive.

Restricting a grammar also changes how a proposal budget is allocated. The locked branch can spend more of its finite candidate budget on non-generator edits. That consequence belongs to this permission intervention; the comparison does not identify the effect of one isolated accepted generator edit with every later candidate held fixed. Neither branch has a claim to represent the best possible search strategy. The result bounds the implemented procedure and its declared resource allocation.

C.2 RESULTS AND INTERPRETATION

The continued-minus-frozen transfer contrast is -0.01080 ($[-0.02201, -0.00264]$). The fixed-meta contrast is -0.00013 ($[-0.00192, 0.00199]$), and the generator-locked contrast is -0.00086 ($[-0.00491, 0.00232]$). The latter intervals span gains and harms. They establish neither equivalence nor a stable benefit from permitting generator edits. Retention changes by 0.00007 ($[7.52 \times 10^{-7}, 0.00020]$), a small adverse effect.

The original advantage over fixed meta and the fresh unresolved contrast limit a general superiority claim. Stronger conclusions would require independent task families, additional fixed optimizers and a predefined practical effect threshold for any equivalence analysis.

C.3 CHRONOLOGY AND RECONSTRUCTION

The original source/protocol hash freeze preceded its designated run. The follow-up hash freeze followed inspection of that run and preceded its own fresh-family execution. Local Git archival

commits were created later and retain their actual timestamps; they are not evidence of pre-run Git registration. The artifact includes both freeze records, the unmodified original source, the extension, raw outputs and an anonymous portable history. This is local execution provenance, not an externally registered study.

Before fresh execution, the three original branches and checkpoint retention reference were reproduced on the development fixtures and matched the original implementation exactly. The lock checker enumerates allowed programs, verifies generator preservation in candidates and descendants, checks fit caps and rejects a corrupted descendant that violates the lock. The final audit separately reconstructs numerical endpoints and paired summaries. These are internal implementation checks; an independent external audit remains a separate activity.

D Powered replications, code–history crossing and non-recursive baselines

D.1 Chronology and protocol

Both original cohorts (main: seeds 8101–8132, $n = 32$; follow-up: seeds 9401–9432, $n = 32$) were completed and inspected before the replication protocol was written. The replication seed lists, cohort sizes, equivalence margin and inference rules were committed to `replication_protocol.json` and hashed in `replication_freeze_manifest.json` (timestamp `2026-09-25T17:18:49Z`). No replication seed had executed at that moment. The freeze records the SHA-256 of every executed source file; `replication_runner.py` verifies agreement at runtime before any seed executes. Only seed substitution and output-directory routing differ between original and replication runs; no logic inside `worldmodel.py` or `followup_meta_lock.py` was altered.

The main-replication cohort uses seeds 8201–8452 ($n = 252$); the follow-up-replication cohort uses seeds 9501–9752 ($n = 252$). The value 252 is a multiple of both the six-mask rotation and the two initial-state signs in the frozen code, ensuring every rotation cell is balanced. The two replication cohorts are never pooled with each other or with either original cohort.

Inference. Task losses are averaged within each family. Paired percentile bootstrap intervals are computed over families (10 000 resamples, seed 99101), matching the routine used for the original cohorts. Two contrasts are pre-declared for equivalence testing: continued minus generator-locked (follow-up replication) and continued minus fixed meta (both replication cohorts). **Equivalence margin** $\delta = 0.005$ MPC transfer loss: approximately 0.55% of the continued-arm mean (≈ 0.909) and one-third of the continued-versus-frozen benefit (≈ 0.014). A difference smaller than one-third of the demonstrated value of continued editing is treated as practically negligible. The two one-sided tests use the 90% paired percentile bootstrap interval.

- *Detected difference*: 95% CI excludes zero.
- *Practically equivalent*: entire 90% CI $\subset [-0.005, +0.005]$.
- *Inconclusive*: neither condition holds.

All replication contrasts are reported regardless of direction.

D.2 Replication contrasts

Table 9 collects all primary replication contrasts, shown alongside the original cohort for each experiment. The B/T/W column counts families with a negative/zero/positive difference; for "continued minus" rows this is continued better/tied/worse. The 90% CI and decision columns are filled only for the equivalence-tested contrasts.

Summary. E1 procedure and structure main effects replicate with the same sign and larger magnitude (procedure share: 45% in the replication vs. 38% in the original). The E2 own-versus-replay interaction replicates ($+0.00987$, 95% CI $[+0.00036, +0.02570]$, $n = 252$; original $+0.00257$, $[+0.00003, +0.00622]$). E3 continued-minus-frozen replicates in both cohorts (-0.01010, main;

Contrast	Mean	95% CI	90% CI	B/T/W	Decision
E1: Decomposition of naive descendant gain					
Procedure main effect (orig., $n = 32$)	-0.00101	$[-0.00271, -0.00002]$		9/20/3	
Procedure main effect (rep., $n = 252$)	-0.00406	$[-0.00936, -0.00087]$		96/120/36	
Structure main effect (orig., $n = 32$)	-0.00165	$[-0.00337, -0.00060]$		23/6/3	
Structure main effect (rep., $n = 252$)	-0.00489	$[-0.01029, -0.00150]$		186/28/38	
Naive diagonal gain (orig., $n = 32$)	-0.00266	$[-0.00605, -0.00066]$		23/6/3	
Naive diagonal gain (rep., $n = 252$)	-0.00895	$[-0.01962, -0.00248]$		185/28/39	
E2: Evidence regime (inherited arm, final round)					
Own evidence: inherited minus base (orig., $n = 32$)	-0.04602	$[-0.09462, -0.00975]$		22/7/3	
Own evidence: inherited minus base (rep., $n = 252$)	-0.06408	$[-0.08648, -0.04333]$		176/46/30	
Replay evidence: inherited minus base (orig., $n = 32$)	-0.04859	$[-0.09760, -0.01133]$		22/8/2	
Replay evidence: inherited minus base (rep., $n = 252$)	-0.07395	$[-0.10041, -0.05028]$		180/54/18	
Own minus replay (interaction) (orig., $n = 32$)	$+0.00257$	$[+0.00003, +0.00622]$		1/27/4	
Own minus replay (interaction) (rep., $n = 252$)	$+0.00987$	$[+0.00036, +0.02570]$		15/203/34	
E3 (main protocol): transfer					
Continued minus frozen (orig., $n = 32$)	-0.01414	$[-0.02527, -0.00462]$		27/2/3	
Continued minus frozen (rep., $n = 252$)	-0.01010	$[-0.01501, -0.00579]$		198/22/32	
Continued minus fixed meta (orig., $n = 32$)	-0.00988	$[-0.01961, -0.00208]$	$[-0.01798, -0.00281]$	11/16/5	
Continued minus fixed meta (rep., $n = 252$)	-0.00129	$[-0.00274, +0.00002]$	$[-0.00248, -0.00019]$	71/139/42	*equiv.*
E3 (follow-up protocol): transfer					
Continued minus frozen (orig., $n = 32$)	-0.01080	$[-0.02201, -0.00264]$		26/1/5	
Continued minus frozen (rep., $n = 252$)	-0.00697	$[-0.01024, -0.00397]$		197/21/34	
Continued minus fixed meta (orig., $n = 32$)	-0.00013	$[-0.00192, +0.00199]$	$[-0.00171, +0.00157]$	8/22/2	
Continued minus fixed meta (rep., $n = 252$)	-0.00425	$[-0.00704, -0.00187]$	$[-0.00655, -0.00221]$	76/140/36	*detected*
Continued minus generator-locked (orig., $n = 32$)	-0.00086	$[-0.00491, +0.00232]$	$[-0.00409, +0.00160]$	4/24/4	
Continued minus generator-locked (rep., $n = 252$)	$+0.00031$	$[-0.00217, +0.00251]$	$[-0.00171, +0.00219]$	49/170/33	*equiv.*
Retention (source singletons; positive = worse)					
Main: continued minus frozen (orig., $n = 32$)	$+0.00073$	$[-7.4 \times 10^{-7}, +0.00218]$		4/25/3	
Main: continued minus frozen (rep., $n = 252$)	$+0.00026$	$[+0.00004, +0.00058]$		23/184/45	
Follow-up: continued minus frozen (orig., $n = 32$)	$+0.00007$	$[+7.5 \times 10^{-7}, +0.00020]$		2/24/6	
Follow-up: continued minus frozen (rep., $n = 252$)	$+0.00047$	$[+0.00007, +0.00104]$		25/185/42	
Code × history crossing (E3 main protocol)					
Code effect (avg. history) (orig., $n = 32$)	-0.01414	$[-0.02526, -0.00461]$		27/2/3	
Code effect (avg. history) (rep., $n = 252$)	-0.01029	$[-0.01510, -0.00615]$		198/22/32	
History effect (avg. code) (orig., $n = 32$)	-1.8×10^{-6}	$[-0.00001, 0]$		1/31/0	
History effect (avg. code) (rep., $n = 252$)	$+0.00019$	$[-0.00002, +0.00060]$		4/247/1	
Code × history interaction (orig., $n = 32$)	-3.7×10^{-6}	$[-0.00001, 0]$		1/31/0	
Code × history interaction (rep., $n = 252$)	-0.00042	$[-0.00124, +4.9 \times 10^{-7}]$		4/247/1	

Table 9: Original and replication contrasts (loss difference, lower is better). B/T/W counts families with a negative/zero/positive difference ($|d| \leq 10^{-12}$ is a tie); for "continued minus" rows this is continued better/tied/worse. The 90% CI and Decision columns apply only to the pre-declared equivalence contrasts ($\delta = 0.005$): *equiv.* if the entire 90% CI lies in $[-0.005, 0.005]$; *detected* if the 95% CI excludes zero; *inconcl.* otherwise. Decisions are stated for the replication cohorts, for which the rule was declared.

-0.00697, follow-up). The fixed-meta contrast is practically equivalent in the main replication (-0.00129, 90% CI $[-0.00248, -0.00019]$) and shows a small but detected advantage for continued editing in the follow-up replication (-0.00425, 90% CI $[-0.00655, -0.00221]$, outside $\pm\delta$ on the lower side). The generator-locked contrast is practically equivalent: $+0.00031$, 90% CI $[-0.00171, +0.00219] \subset [-0.005, +0.005]$. Retention shows small but detected adverse changes in both replication cohorts: $+0.00026$ ($[+0.000038, +0.000585]$, main) and $+0.00047$ ($[+0.000068, +0.001039]$, follow-up).

D.3 Code × history crossing

Method. The crossing analysis re-executes the frozen source-training and E3 continuation logic of `worldmodel.py` (same functions, same seeds, same task seeds). After continuation, each family holds three final programs (continued, frozen, fixed-meta) and three final histories. The script evaluates every combination of code arm and history arm on the held-out transfer tasks and asserts that every *diagonal* cell reproduces the saved outcome—96 diagonal (family, arm) cells for the original cohort and 756 for the main-replication cohort, each checked task by task on its four transfer tasks with absolute tolerance 10^{-12}. All assertions pass, so the post-hoc re-execution reproduces the original runs. Off-diagonal cells reveal whether the package effect is carried by the program, by the history, or by an interaction of the two.

Results. Table 10 shows mean transfer loss for the 2×2 code-arm (continued/frozen) × history-arm (continued/frozen) factorial. The code effect is defined as the code contrast averaged over both history arms; the history effect and interaction are analogously defined.

		Original ($n = 32$)		Main rep. ($n = 252$)	
Code arm	History arm	Mean	Δ vs. fro./fro.	Mean	Δ vs. fro./fro.
Continued	Continued	0.90934	−0.01414	0.84521	−0.01010
Continued	Frozen	0.90934	−0.01414	0.84523	−0.01008
Frozen	Continued	0.92347	0	0.85570	+0.00040
Frozen	Frozen	0.92347	—	0.85530	—
Code effect (avg. history)		-0.01414 $[-0.02526, -0.00461]$		-0.01029 $[-0.01510, -0.00615]$	
History effect (avg. code)		-1.8×10^{-6} $[-0.00001, 0]$		$+0.00019$ $[-0.00002, +0.00060]$	
Code × history interaction		-3.7×10^{-6} $[-0.00001, 0]$		-0.00042 $[-0.00124, +4.9 \times 10^{-7}]$	

Table 10: Code × history factorial (mean transfer loss, lower is better; effects with 95% CI). A history is the complete record an arm carries at the end of continuation: the shared source history, the continuation-stage evidence, the arm's model-repair records and, for the continued arm, its procedure-edit records. The frozen arm's history therefore contains continuation evidence and repair records but no procedure edits. Diagonal cells reproduce the saved outcomes (asserted per task, absolute tolerance 10^{-12}).

Interpretation. In the original cohort ($n = 32$) the history effect is exactly zero in 31/32 families and the interaction is of order 10^{-6}. The continued code achieves its lower loss when paired with the frozen arm's history, and the frozen code gains nothing from the continued arm's history. In the main-replication cohort ($n = 252$) the history effect is $+0.00019$ (95% CI $[-0.00002, +0.00060]$) with 247/252 families exactly tied; the few non-tied families make continued history slightly worse for the frozen code. It is two orders of magnitude smaller than the code effect (-0.01029). **The benefit of continued editing is a code effect in both cohorts.**

The frozen checkpoint uses `history/single` repair in all families of both cohorts, reflecting the single-mechanism source curriculum. Continued editing moves the majority of families to `history/paired` in both cohorts (21/32 original; 145/252 replication; see Table 12).

D.4 Evaluation-time reduction to six behaviors

Only the diagnosis and repair subtrees execute during a transfer repair; the acquisition and meta subtrees are inactive. Across the full grammar of 54 programs (3 diagnosis × 2 repair × 3 acquisition

$\times$ 3 meta variants), this reduces to $3 \times 2 = 6$ evaluation-time behaviors (diagnosis $\in$ {cyclic, history, residual} $\times$ repair $\in$ {paired, single}). The crossing script checks this reduction directly: for the first four families of each cohort and for every history, it evaluates each behavior with two programs that differ in both inactive subtrees (`passive`/`cycle` versus `goal_disagreement`/`compose`) and asserts identical transfer and validation losses. A behavior can still depend on the history it is given, through the `history` diagnosis; Table 11 reports each behavior under the checkpoint history and under the continued arm's final history.

Cohort	History	cyc./paired	cyc./single	hist./paired	hist./single	resid./paired	resid./single
Original ($n = 32$)	Checkpoint	0.91642	0.92705	0.90827	0.92347	0.90798	0.92347
Original ($n = 32$)	Continued	0.91642	0.92705	0.90826	0.92347	0.90798	0.92347
Main rep. ($n = 252$)	Checkpoint	0.84960	0.85674	0.84554	0.85570	0.84393	0.85530
Main rep. ($n = 252$)	Continued	0.84960	0.85674	0.84551	0.85570	0.84393	0.85530

Table 11: Mean transfer loss for the six evaluation-time behaviors, averaged over families within each cohort, for checkpoint and continued histories. The `cyclic` and `residual` behaviors do not read the history and are identical under both; the `history` behaviors differ by at most 3×10^{-5} in cohort means. Lower is better. The best single fixed behavior is `residual/paired` in both cohorts.

Arm	hist./single	hist./paired	resid./paired	resid./single	cyc./paired	cyc./single
Original ($n = 32$)						
Frozen	32	—	—	—	—	—
Continued	—	21	6	2	3	—
Fixed meta	10	8	10	1	3	—
Main replication ($n = 252$)						
Frozen	252	—	—	—	—	—
Continued	10	145	64	12	21	—
Fixed meta	61	85	72	9	25	—

Table 12: Final evaluation-time behavior counts per arm, by cohort. The frozen checkpoint is `history/single` in every family of both cohorts. Continued editing moves the majority of families to `history/paired` (21/32 original; 145/252 replication). The fixed-meta arm selects among five of the six behaviors in both cohorts.

Non-recursive baselines. We evaluate three reference methods that operate over the six evaluation-time behaviors only:

1. **Validation-selected (6-way)**: for each transfer task, select among the six behaviors the one whose repaired model has the lowest error on that task's validation split, each behavior using the same 3-fit repair budget as the lineage's own repair. This requires $6 \times 3 = 18$ fits per task versus 3 for the main protocol. *Note: choosing the best fixed behavior by its test mean would be optimistic; this baseline uses held-out validation loss, not test loss.*
2. **Uniform-random behavior**: the expected loss of choosing one of the six behaviors uniformly at random per task, i.e. the average over the six behaviors.
3. **Test-oracle ceiling**: for each task, the behavior with the lowest test loss. This is *not a deployable method*; it serves as an upper reference on the value of perfect behavior selection.

Original cohort ($n = 32$). Continued minus validation-selected: $+0.00411$, 95% CI $[-0.00009, +0.01001]$ (90%: $[+0.00050, +0.00891]$), 7/8/17 better/tie/worse. Continued is *worse* than validation-selected in 17/32 families. Continued minus test-oracle: $+0.00655$, 95% CI $[+0.00186, +0.01257]$; continued is worse in 28/32 families relative to the oracle ceiling. Continued minus uniform-random: -0.00844, 95% CI $[-0.01521, -0.00217]$; continued outperforms random in 25/32 families.

Main replication cohort ($n = 252$). Continued minus validation-selected: $+0.00262$, 95% CI $[+0.00026, +0.00528]$, 78/46/128 better/tie/worse: continued is detectably worse than validation

selection. Continued minus test-oracle: $+0.00511$, 95% CI $[+0.00295, +0.00765]$. Continued minus uniform-random: -0.00592, 95% CI $[-0.00866, -0.00345]$; continued outperforms the random average in 208/252 families.

In both cohorts continued editing beats the uniform-random average but is not better than selecting among the six fixed behaviors by each task's validation error (original: interval includes zero; replication: continued detectably worse). Validation selection uses six times as many fits per task at evaluation (18 versus 3) but needs no continuation stages.

D.5 Robustness

Table 13 supplements the paired summaries with the median, sign counts and the leave-one-family-out (LOO) range (minimum and maximum of the means obtained by deleting one family at a time). Deleting any single family never changes the sign of a listed mean whose 95% interval excludes zero. Single families do carry weight in some replication means: the E2 interaction ranges over $[+0.00347, +0.00992]$ and the naive gain over $[-0.00918, -0.00461]$.

Contrast (cohort)	n	Median	B/T/W	LOO min	LOO max
E1					
Naive diagonal gain (orig.)	32	-0.00066	23/6/3	-0.00285	-0.00113
Naive diagonal gain (rep.)	252	-0.00057	185/28/39	-0.00918	-0.00461
Procedure main effect (orig.)	32	0	9/20/3	-0.00110	-0.00024
Procedure main effect (rep.)	252	0	96/120/36	-0.00412	-0.00190
Structure main effect (orig.)	32	-0.00068	23/6/3	-0.00175	-0.00089
Structure main effect (rep.)	252	-0.00045	186/28/38	-0.00510	-0.00271
E2					
Own minus replay (orig.)	32	0	1/27/4	$+0.00130$	$+0.00265$
Own minus replay (rep.)	252	0	15/203/34	$+0.00347$	$+0.00992$
E3 transfer (main protocol)					
Cont. minus frozen (orig.)	32	-0.00088	27/2/3	-0.01539	-0.01147
Cont. minus frozen (rep.)	252	-0.00059	198/22/32	-0.01079	-0.00886
Cont. minus fixed meta (orig.)	32	0	11/16/5	-0.01047	-0.00707
Cont. minus fixed meta (rep.)	252	0	71/139/42	-0.00154	-0.00097
E3 transfer (follow-up protocol)					
Cont. minus frozen (orig.)	32	-0.00051	26/1/5	-0.01164	-0.00633
Cont. minus frozen (rep.)	252	-0.00058	197/21/34	-0.00744	-0.00620
Cont. minus fixed meta (orig.)	32	0	8/22/2	-0.00096	$+0.00036$
Cont. minus fixed meta (rep.)	252	0	76/140/36	-0.00449	-0.00347
Cont. minus gen.-locked (orig.)	32	0	4/24/4	-0.00172	$+0.00079$
Cont. minus gen.-locked (rep.)	252	0	49/170/33	-0.00012	$+0.00112$
Code × history crossing and baselines					
Code effect (orig.)	32	-0.00088	27/2/3	-0.01539	-0.01147
Code effect (rep.)	252	-0.00059	198/22/32	-0.01079	-0.00905
History effect (orig.)	32	0	1/31/0	-1.9×10^{-6}	0
History effect (rep.)	252	0	4/247/1	-0.00001	$+0.00020$
Cont. minus val.-selected (orig.)	32	$+0.00001$	7/8/17	$+0.00192$	$+0.00440$
Cont. minus val.-selected (rep.)	252	$+0.00001$	78/46/128	$+0.00197$	$+0.00309$

Table 13: Robustness diagnostics: median family difference, sign counts (negative/zero/positive, tie if $|d| \leq 10^{-12}$) and the leave-one-family-out (LOO) range, i.e. the minimum and maximum of the n means obtained by deleting one family at a time.

D.6 Robustness to an unrepresentable dynamics term

Design. In the frozen testbed the simulator and the model share one function dictionary, so every world lies in the model class. To test whether the ladder's verdicts depend on this realizability, a post-hoc cohort of 252 fresh families (seeds 8601–8852, disjoint from all other cohorts) adds to the

simulator's velocity update a term the dictionary cannot represent:

$$v'_{\text{world}} = v'_{\text{frozen}} + \kappa_f \sin(2.5\,x + \psi_{f,m})\,v, \qquad \kappa_f \sim U(0.20, 0.35), \quad \psi_{f,m} \sim U(0, 2\pi),$$

with one κ_f per family (cohort mean 0.277) and one phase per family and mask, so the term changes across contexts. Both are drawn from dedicated generators that consume none of the frozen random streams; the position update keeps its frozen form. Only `World.step` is replaced. The dictionary, fitting, repair, procedures, planner and evaluator are the frozen code, and every data-generating call (`sampled_data`, `acquire_trajectory`) and every evaluation rollout (`evaluate`) uses the modified world, while the planner plans with the learned (now misspecified) model. Re-executing two families with the frozen `run_instance` and the patch reproduces their saved outcomes exactly; without the patch all 81 outcomes of a family change.

Calibration. The range of κ_f was set on development seeds 101–103 only (all 14 masks, 2000 transitions per world, model fitted at the true mask). There the term carries on average 9.4% of the variance of the velocity-residual target the dictionary must explain (range 2.9%–33% across worlds), and 98% of its variance remains in the residual of the true-mask fit. A full-pipeline timing check on the same development seeds followed; κ was not changed after it.

Protocol. The term, seeds, reported contrasts and a directional prediction (continued editing below frozen on transfer; procedure share below 100% of the naive gain; no success threshold) were written to `misspec_protocol.json` and hashed, with the runner and the frozen code, in `misspec_freeze_manifest.json` at `2026-09-25T17:32:36Z`, before any designated seed ran. One amendment preceded execution and is recorded in the protocol: a first freeze (`17:31:40Z`) called `worldmodel.main()`, which stopped before any seed because the original freeze manifest also hashes `README.md`, a documentation file edited after the original freeze and never executed. The runner was changed to verify every original code hash except `README.md` and to run the same designated loop itself (semantic checks, one `run_instance` per seed, the frozen aggregation), then re-frozen. `README.md` has since been restored to its frozen bytes, and all original hashes match. Inference is unchanged (paired percentile bootstrap, 10 000 resamples, seed 99101; $\delta = 0.005$ for continued minus fixed meta). The cohort is never pooled.

Contrast	Misspecified	95% CI	B/T/W	Main rep. (ref.)
E1 naive descendant gain	-0.01291	$[-0.02152, -0.00515]$	168/26/58	-0.00895
E1 procedure main effect	-0.00491	$[-0.00858, -0.00198]$	88/123/41	-0.00406
E1 structure main effect	-0.00800	$[-0.01375, -0.00224]$	165/25/62	-0.00489
procedure share of naive gain	38%			45%
fixed meta on base model, share	62%			62%
E2 own minus replay (inherited)	0.01672	$[0.00568, 0.03002]$	61/115/76	0.00987
E3 transfer: continued minus frozen	-0.00757	$[-0.01264, -0.00237]$	161/32/59	-0.01010
E3 transfer: continued minus fixed meta	0.00128	$[-0.00261, 0.00577]$	65/135/52	-0.00129
90% CI		$[-0.00200, 0.00501]$		
E3 retention: continued minus frozen	0.00057	$[-0.00018, 0.00175]$	83/73/96	0.00026

Table 14: Misspecified cohort ($n = 252$, seeds 8601–8852; loss differences, lower is better; B/T/W as in Table 9). The last column repeats the main-replication estimate for orientation; the cohorts use different seeds and are not compared statistically. Mean transfer loss is higher by a factor of about 1.042 (continued: 0.8809 versus 0.8452).

Interpretation. The qualitative pattern persists (Table 14). The inherited procedure accounts for 38% of the naive descendant gain, the rest being inherited structure, and a fixed meta-procedure applied to the base model recovers 62%. Donor replay again overstates the inherited procedure's benefit relative to owned evidence (interaction 0.01672, interval excluding zero). Continued editing beats freezing on transfer (-0.00757, $[-0.01264, -0.00237]$). Continued minus fixed meta is inconclusive: the point estimate slightly favors the fixed meta-procedure, the 95% interval spans zero, and the upper 90% bound (0.00501) lies just outside δ, so equivalence is not declared. The adverse retention change is not detected here (0.00057, $[-0.00018, 0.00175]$). This tests one form of misspecification (a smooth state-dependent velocity product with one κ range) added to a model class whose symbolic support still matches the worlds' mask structure;

other unmodeled forms, such as time-varying dynamics or state-dependent noise, were not tested.

E REGISTRATION, MECHANISM TESTS AND STATISTICAL RE-ANALYSIS

This appendix documents every cohort and analysis in the paper, the three mechanism tests that followed the replication (curriculum, positive control, independent source lineages), a lifetime-cost analysis, and a statistical re-analysis of all saved outcomes. Every cohort is analysed separately; no two cohorts are pooled. The sign convention throughout is first-named minus second-named arm in MPC loss, so a negative value favours the first-named arm. Unless stated otherwise, inference is the paired percentile bootstrap over families (10 000 resamples, `default_rng(99101)`), with 95% and 90% intervals and the equivalence margin $\delta = 0.005$ declared in `replication_protocol.json`. We apply the margin to every contrast, not only to the two for which it was declared.

E.1 REGISTRATION STATUS

Table 15 lists each experiment and analysis with its seeds, protocol file, freeze time and status. *Pre-declared* means that the protocol, the executed code and the seed list were hashed in a freeze manifest before any designated seed ran; the runners verify these hashes at start-up. Only the original main cohort was designed before any outcome was seen. Every later protocol was written after at least one earlier cohort had been opened, so its design is post-hoc even where its predictions and analysis were fixed before execution. The replication protocol in particular was declared before its seeds ran but after both original cohorts were inspected.

Three departures from a clean prospective sequence are recorded in the protocol or summary files. (i) *Misspecification.* A first freeze (17:31:40) aborted before executing any seed, because the original manifest also hashes a documentation file that had been edited; the runner was changed to skip that file's hash and re-frozen (Appendix D.6). (ii) *Positive control.* Before its protocol was written, the code path was run on 16 pilot seeds (301–316, never designated in any cohort) and the pilot means were inspected. The history-ablation cells were added after the first pilot pass and the predictions were written after the pilot; $K = 2$ and the other design choices were fixed before it. Pilot seeds are excluded from every analysis. (iii) *Independent lineages.* One line of the analysis script was changed after the designated run (amendment A1, recorded with the frozen script hash): the descriptive "frozen without continuation" loss in the lifetime analysis uses each run's own checkpoint behavior rather than a hard-coded `history/single`. This changes only descriptive values under separate histories; no primary contrast is affected.

The code×history crossing, the six-behavior baselines (validation-selected, oracle, uniform random), the compute-matched baseline and the round-3 statistics are analyses of saved outcomes that no protocol declared; so is the independent verification of the lineage experiment (full re-execution and a history-length placeholder ablation, Appendix E.4). The first three re-execute the frozen code and reproduce the saved outcomes exactly before computing anything new; the compute-matched baseline's selection rule was fixed in the script's docstring before its outcomes were computed but was not hashed.

E.2 CURRICULUM TEST

Hypothesis and intervention. In every earlier cohort the frozen checkpoint uses the `history/single` behavior. A natural explanation is that the source curriculum contains only single-mechanism tasks (source masks $[1, 2, 4, 8]$), so the inherited procedure specializes to single-term repair and continued editing gains by escaping this specialization. The test changes only the list of source-stage masks, from ORIGINAL $[1, 2, 4, 8]$ to MIXED $[1, 3, 4, 12]$. The stage rule (tasks $C[s]$ and $C[(s + 1) \bmod 4]$) is kept, so every MIXED stage contains one single-mechanism task and one pair task, and all four primitives remain covered. Stages, tasks per stage, observations, task seeds, candidate budget, E3 continuation, transfer and retention tasks and fit caps are unchanged; realized fits are identical in both curricula (252 source fits and 192 E3 fits per arm). The design is paired: each of the 252 families (seeds 8901–9152) runs both curricula. The ORIGINAL arm, run

Table 15: Registration status of every experiment and analysis (all times UTC on 2026-09-25). *Pre-declared*: protocol and code hashed in a freeze manifest before any designated seed executed. *Post-hoc design*: the protocol was written after earlier cohorts had been opened. Rows without a freeze time are analyses of saved outcomes that no protocol declared.

Experiment	Seeds	n	Protocol / script	Freeze	Status	Deviations, amendments
Original main (E1–E3)	8101–8132	32	protocol.json	15:17:09	pre-declared (prospective)	none; discovery cohort
Follow-up (generator lock)	9401–9432	32	followup_ protocol.json	15:25:38	pre-declared; designed after main outcomes were opened	none
Main replication	8201–8452	252	replication_ protocol.json	17:18:49	pre-declared; designed after both original cohorts were opened	none
Follow-up replication	9501–9752	252	replication_ protocol.json	17:18:49	pre-declared (as above)	none
Misspecification	8601–8852	252	misspec_ protocol.json	17:32:36	pre-declared; post-hoc design	one pre-execution amendment: first freeze (17:31:40) aborted before any seed (README hash); runner re-hashed and re-frozen
Curriculum test	8901–9152	252	curriculum_ protocol.json	17:59:23	pre-declared (P1–P3); post-hoc design	search-path trace (30 families) exploratory
Positive control	9801–10052	252	posctrl_ protocol.json	18:08:53	pre-declared after a pilot; post-hoc design	16 pilot seeds (301–316) inspected before predictions; history-ablation cells added after first pilot pass
Independent lineages	10101–10226	126×4	lineage_ protocol.json	18:14:24	pre-declared (P1–P3, E1–E6); post-hoc design	1 post-run one-line analysis change (A1): descriptive no-continuation loss; no primary affected
Lifetime cost	10101–10226; saved cohorts	126; 252; 32	lineage_ protocol.json	18:14:24	declared as post-hoc descriptive; not primary	as lineages
Lineage verification (re-execution, placeholder ablation)	10101–10226	126×4	lineage_runner.py (re-run); verifier scripts	—	post-hoc, undeclared diagnostic	rerun identical in every cell except wall time
Code×history crossing; behavior baselines	8101–8132; 8201–8452	32; 252	extension_e3_ cross.py	—	post-hoc, undeclared	reproduces saved diagonal exactly
Compute-matched baseline	8101–8132; 8201–8452	32; 252	extension_ matched_ baseline.py	—	post-hoc, undeclared (rule fixed in docstring before outcomes; not hashed)	none
Round-3 statistics	all saved cohorts	—	analysis_round3.py	—	post-hoc, undeclared	Holm families declared in script

through the new code path, reproduces the saved replication outcomes of seeds 8201–8203 exactly (tolerance 10^{-12}), before the freeze and again inside the designated run.

Predictions. P1: the share of frozen checkpoints with single-term repair drops under MIXED. P2: the continued-minus-frozen gap shrinks, i.e. the difference-in-differences (DiD) is positive. P3: frozen transfer loss decreases under MIXED. No success threshold was set.

Results. None of the predictions is supported (Tables 16 and 17). The frozen checkpoint is `history/single` in 252/252 families under ORIGINAL and 252/252 under MIXED, and no family switches behavior (P1). The DiD is -0.00029 ($[-0.00114, +0.00036]$), with 221 exact ties; the gap is -0.00687 under ORIGINAL and -0.00717 under MIXED (P2). Frozen transfer loss is identical in 251 of 252 families (P3). Continued editing reaches paired repair in 93.7% (ORIGINAL) and 94.4% (MIXED) of families. Retention effects are all below 3×10^{-4} in magnitude. The curriculum-content explanation is therefore refuted as stated; the continued-over-frozen gap is insensitive to curriculum content.

Table 16: Curriculum test ($n = 252$ families, paired within family). Transfer or retention MPC loss; negative = first term better. P2 and P3 are the pre-declared paired contrasts.

Contrast	Mean	95% CI	90% CI	−/tie/+	Verdict ($\delta = 0.005$)
Transfer, pre-declared					
P2: DiD (continued−frozen), MIXED−ORIGINAL	−0.00029	[−0.00114, +0.00036]	[−0.00096, +0.00028]	15/221/16	equivalent
P3: frozen, MIXED−ORIGINAL	−0.00007	[−0.00022, 0.00000]	[−0.00022, 0.00000]	1/251/0	equivalent
Transfer, reported					
continued−frozen, ORIGINAL	−0.00687	[−0.01040, −0.00353]	[−0.00980, −0.00406]	203/16/33	det., not equiv.
continued−frozen, MIXED	−0.00717	[−0.01076, −0.00377]	[−0.01009, −0.00431]	206/13/33	det., not equiv.
continued, MIXED−ORIGINAL	−0.00037	[−0.00118, +0.00027]	[−0.00103, +0.00018]	15/222/15	equivalent
fixed meta, MIXED−ORIGINAL	−0.00145	[−0.00309, 0.00000]	[−0.00280, −0.00020]	54/161/37	equivalent
continued−fixed meta, ORIGINAL	+0.00064	[−0.00129, +0.00311]	[−0.00104, +0.00267]	58/156/38	equivalent
continued−fixed meta, MIXED	+0.00173	[−0.00042, +0.00431]	[−0.00011, +0.00384]	41/169/42	equivalent
DiD (continued−fixed meta), MIXED−ORIGINAL	+0.00109	[−0.00054, +0.00283]	[−0.00026, +0.00254]	44/145/63	equivalent
Retention (source singletons)					
continued−frozen, ORIGINAL	−0.00008	[−0.00029, +0.00004]	[−0.00026, +0.00004]	29/179/44	equivalent
continued−frozen, MIXED	+0.00002	[−0.00004, +0.00006]	[−0.00003, +0.00005]	28/178/46	equivalent
fixed meta change vs checkpoint, ORIGINAL	+0.00010	[+0.00004, +0.00021]	[+0.00004, +0.00020]	23/182/47	det., equiv.
checkpoint, MIXED−ORIGINAL	0.00000	[0.00000, 0.00000]	[0.00000, 0.00000]	0/251/1	equivalent

Verdict: *det.* = 95% percentile interval excludes 0; *equiv.* = 90% interval inside $\pm\delta$; *not equiv.* = detected, 90% interval not inside $\pm\delta$; *relevant* = detected, 90% interval entirely outside $\pm\delta$; *not detected* = neither.

Table 17: Curriculum test: evaluation-time behaviors (families out of 252). *Source fixed meta* is the fixed-meta lineage at the end of source training.

	ORIGINAL		MIXED	
Program	history/single	paired	history/single	paired
Checkpoint (= frozen)	252	0	252	0
Continued	4	236	3	238
E3 fixed meta	44	200	27	225
Source fixed meta	43	168	55	188

Exploratory search-path trace (not pre-declared). To locate the lock-in, we traced the inherited source lineage's proposals and accepted edits in 30 families of each curriculum (MIXED: seeds 8901–8930; ORIGINAL: the raw logs of replication seeds 8201–8230). The accepted edit path is nearly the same in every family: no acceptance at stages 0 and 1 (occasionally a diagnosis change to `residual`), a meta edit at stage 2, and a diagnosis change to `history` at stage 3. A `repair:single→paired` edit is proposed only at stages 2–3. It has positive procedure gain in 33 of its 60 proposals under MIXED (29 of 60 under ORIGINAL) and is never accepted, because a different candidate with larger gain wins and only one edit is accepted per stage. Under the same singleton curriculum the fixed-meta source lineage reaches paired repair in 168/252 families (188/252 under MIXED; Table 17). This trace suggested a search-path explanation within the four-stage

horizon. The independent-lineage experiment (Appendix E.4) locates the cause more precisely: the proposal window of the base generator depends on the history length that the shared-history contract produces.

E.3 POSITIVE CONTROL

Question. Only the diagnosis and repair subtrees execute when a final program repairs a model at evaluation, so the 54 programs collapse to six behaviors, and the generator and the history are inert at test time. The generator-lock and history nulls could therefore be guaranteed by design. The positive control asks whether the same contrasts, cohort size and inference detect generator and history value when these components do execute.

Design. *K0* is the paper's evaluation regime: on each transfer task the arm's final program performs one protected 3-fit repair, followed by the unchanged MPC evaluation. *TTSE* (test-time self-editing, $K = 2$) makes the generator and the history execute at evaluation. On each of the four transfer tasks, independently and without carry-over, the arm starts from its final E3 program and history, appends the task's evidence record, and runs two rounds of the frozen procedure-editing step (`worldmodel.evolve`, or `followup_meta_lock.evolve_locked` for the locked generator) with the frozen candidate budget of 4 and 3-fit repairs. The scorer sees only that task's own train/validation split; the protected test rollouts are never visible. The protected repair and MPC evaluation then follow as in K0. Grammar, scorer, budgets, tasks, checkpoints and evaluator are unchanged. Source training and E3 for the arms continued, frozen, fixed meta and generator-locked are the frozen follow-up functions.

Arms and planted effects. Besides the four natural arms, two cells have known ground truth. *Inactive-only* (generator rung) keeps continued's code and history but locks the test-time generator to one that proposes only acquisition and meta edits, restricted before selection. Such edits never execute during a repair, so this cell equals K0 continued exactly (asserted in all families); it differs from TTSE continued only in how useful the generator is. *History-empty* (history rung) keeps each arm's code and replaces its history by the empty list. The natural code×history crossing of Appendix D.3 is also run in both regimes.

Budgets and reproduction. Every TTSE cell uses exactly 240 edit fits and 12 repair fits per family; K0 cells use 12 repair fits (Table 19). K0 run through the positive-control code reproduces the saved per-task transfer losses of all four arms of the follow-up replication for seeds 9501–9503 exactly (tolerance 10^{-12}), before the freeze and again inside the designated run of 252 fresh families (seeds 9801–10052).

Results. Table 18 reports every declared contrast. The generator-lock rung detects the planted useless generator (PC1, -0.00180, 95% CI $[-0.00326, -0.00061]$), and the history rung detects the planted history ablation under TTSE but not under K0 (PC3). The crossing's history effect, exactly zero in 240 of 252 families under K0, is detected under TTSE, with 79 ties. Under K0 the natural generator lock reproduces the paper's equivalence in a fresh cohort; under TTSE it is not detected and not shown to be equivalent, so natural generator plasticity remains open in this regime. Continued and generator-locked final programs differ in the meta subtree in 50.8% of families and in the full program in 81.7%. Once every arm may edit at test time, continued editing's advantages shrink: the DiD for continued minus frozen is positive and continued minus fixed meta becomes equivalent.

Caveat on the margin. The planted generator effect is detected, yet its 90% interval ($[-0.00300, -0.00077]$) lies inside $\pm\delta$. The declared secondary prediction for PC1 (not practically equivalent) therefore fails, and an effect of this known-real size would also pass our equivalence test. Every "practically equivalent" verdict in the paper bounds the magnitude of an effect at $|\text{effect}| < \delta$; it does not assert that the effect is absent. Under K0 the half-width of the 95% interval of the generator-lock contrast is about 0.0014, which indicates the smallest effect this rung can detect there.

Table 18: Positive control ($n = 252$ families). K0 = frozen evaluation regime of the paper; TTSE = test-time self-editing ($K = 2$). Negative = first term better. PC1 and PC3 are planted effects with known ground truth.

Contrast	Mean	95% CI	90% CI	−/tie/+	Verdict ($\delta = 0.005$)
Generator rung					
PC1 planted: continued−inactive-only (TTSE)	-0.00180	$[-0.00326, -0.00061]$	$[-0.00300, -0.00077]$	54/173/25	det., equiv.
PC2 natural lock: continued−locked (K0)	-0.00013	$[-0.00164, +0.00123]$	$[-0.00136, +0.00103]$	37/175/40	equivalent
PC2 natural lock: continued−locked (TTSE)	-0.00260	$[-0.00622, +0.00044]$	$[-0.00546, 0.00000]$	80/125/47	not detected
PC2 DiD (TTSE−K0)	-0.00247	$[-0.00624, +0.00083]$	$[-0.00552, +0.00032]$	92/88/72	not detected
History rung					
PC3 planted: arm history−empty (K0)	$+0.00012$	$[-0.00001, +0.00031]$	$[0.00000, +0.00027]$	15/216/21	equivalent
PC3 planted: arm history−empty (TTSE)	-0.00369	$[-0.00621, -0.00132]$	$[-0.00573, -0.00170]$	193/0/59	det., not equiv.
Code×history crossing					
PC4 history effect (K0)	$+0.00007$	$[-0.00019, +0.00038]$	$[-0.00013, +0.00031]$	4/240/8	equivalent
PC4 history effect (TTSE)	-0.00172	$[-0.00349, -0.00026]$	$[-0.00318, -0.00046]$	102/79/71	det., equiv.
PC4 code effect (K0)	-0.00996	$[-0.01458, -0.00591]$	$[-0.01378, -0.00648]$	191/18/43	det., relevant
PC4 code effect (TTSE)	-0.00191	$[-0.00396, +0.00006]$	$[-0.00360, -0.00026]$	106/87/59	equivalent
PC4 interaction (K0)	-0.00009	$[-0.00065, +0.00039]$	$[-0.00052, +0.00027]$	6/240/6	equivalent
PC4 interaction (TTSE)	$+0.00494$	$[+0.00220, +0.00836]$	$[+0.00255, +0.00771]$	55/90/107	det., not equiv.
Code rung					
PC5 continued−frozen (K0)	-0.00989	$[-0.01447, -0.00585]$	$[-0.01370, -0.00642]$	188/18/46	det., relevant
PC5 continued−frozen (TTSE)	-0.00363	$[-0.00728, -0.00042]$	$[-0.00663, -0.00091]$	102/89/61	det., not equiv.
PC5 DiD continued−frozen (TTSE−K0)	$+0.00627$	$[+0.00272, +0.01032]$	$[+0.00324, +0.00963]$	58/10/184	det., not equiv.
PC5 continued−fixed meta (K0)	-0.00228	$[-0.00541, -0.00005]$	$[-0.00474, -0.00028]$	67/140/45	det., equiv.
PC5 continued−fixed meta (TTSE)	$+0.00025$	$[-0.00174, +0.00248]$	$[-0.00143, +0.00212]$	92/79/81	equivalent
Value of test-time editing (TTSE−K0, same arm)					
continued	-0.00180	$[-0.00326, -0.00061]$	$[-0.00300, -0.00077]$	54/173/25	det., equiv.
generator-locked	$+0.00068$	$[-0.00275, +0.00451]$	$[-0.00225, +0.00380]$	95/73/84	equivalent
frozen	-0.00806	$[-0.01255, -0.00426]$	$[-0.01178, -0.00480]$	197/0/55	det., not equiv.
fixed meta	-0.00432	$[-0.00805, -0.00144]$	$[-0.00735, -0.00181]$	100/79/73	det., not equiv.

Verdict: *det.* = 95% percentile interval excludes 0; *equiv.* = 90% interval inside $\pm\delta$; *not equiv.* = detected, 90% interval not inside $\pm\delta$; *relevant* = detected, 90% interval entirely outside $\pm\delta$; *not detected* = neither.

Table 19: Positive control: mean transfer loss and realized fits per family (identical budgets within a regime).

	K0		TTSE		
Arm	loss	repair fits	loss	edit + repair fits	accepted edits
Continued	0.87680	12	0.87500	240 + 12	2.79
Generator-locked	0.87692	12	0.87760	240 + 12	2.99
Frozen	0.88669	12	0.87863	240 + 12	4.86
Fixed meta	0.87908	12	0.87476	240 + 12	1.97
Inactive-only (planted)	—	—	0.87680	240 + 12	0.00

E.4 INDEPENDENT SOURCE LINEAGES

Question. Each earlier family has one source lineage, and within it the inherited and fixed-meta source lineages see the same stage tasks and write into one shared history. The test asks whether the E3 conclusions depend on a single source lineage and on this shared-history contract.

Design. 126 fresh families (seeds 10101–10226) each run $L = 4$ independent source lineages that differ only in the task seeds of the source stages; world, masks, grammar, budgets, continuation and transfer tasks are identical across lineages within a family. Lineage 0 reproduces the frozen design exactly. The second factor is the history contract. Under *shared* history (the contract of every earlier cohort) the inherited and fixed-meta source lineages see the same stage tasks and share one history: besides the stage evidence and the inherited lineage's own candidate and repair records, the fixed-meta lineage's candidate records (four per stage) are appended to it. The inherited checkpoint and all E3 arms then continue from this history. Under *separate* histories the fixed-meta source lineage receives its own evidence stream and keeps its own history, and no record crosses between the two source histories; the inherited checkpoint and the E3 arms continue from the inherited-only history. The E3 arms and the six-behavior baselines are those of the frozen E3 code and the crossing script. Realized source fits (both source lineages together) are 252 under shared and 264 under separate histories, because the separate fixed-meta lineage also repairs its own model; the source stage has no fit cap, so this does not change any decision. E3 budgets are identical (Table 21). Shared-history lineage 0 reproduces the saved main-replication outcomes of seeds 8201–8203 exactly (per-task transfer of all arms, final behaviors and every behavior-baseline mean). An independent verification re-executed all 1008 runs (126 families $\times$ 4 lineages $\times$ 2 factors) from scratch; the per-run output agrees with the saved file in every cell except wall time (0 differing cells over 44 columns). It also confirmed that amendment A1 changes one functional line and only the descriptive separate-history "frozen without continuation" losses of the lifetime analysis (Appendix E.5); the primary results are identical under the frozen and amended analysis scripts.

Hierarchical aggregation. For each contrast the family value is the mean over its 4 lineages; the bootstrap resamples the 126 families, so lineages stay nested in their family. Holm adjustment covers the six primary tests (P1 continued minus frozen, P2 continued minus fixed meta, P3 continued minus validation-selected, each under both factors). We also report lineage-0-only estimates, per-lineage counts and the between-lineage share of variance. The lineages vary only in source evidence and their outcomes are highly redundant: continued editing's transfer loss is identical across all 4 lineages in 64/126 families under shared and 100/126 under separate histories, because the transfer tasks are shared, repair starts from the empty model and behaviors that do not read the history ignore it. The inferential unit is therefore the family ($n = 126$); the 504 lineages per factor are not independent draws.

Results. Table 20 gives the full results. Under shared history the E3 pattern replicates: continued minus frozen is -0.00902 ($[-0.01292, -0.00543]$) and every one of the 504 checkpoints is `history/single`. Under separate histories continued minus frozen is -0.00121 ($[-0.00316, +0.00121]$), not detected and practically equivalent, and only 75/504 checkpoints (14.9%) are `history/single`; 365 are `residual/paired`. The factor difference in P1 is $+0.00782$ ($[+0.00404, +0.01258]$). The change comes almost entirely from the frozen arm, whose loss falls by -0.00831 ($[-0.01178, -0.00508]$) under separate histories, while the continued arm changes by -0.00049 ($[-0.00239, +0.00192]$). P2 is detected under shared history (-0.00302, $[-0.00550, -0.00079]$) and not under separate histories. P3 is detected in both regimes and survives Holm: per-task validation selection among the six fixed behaviors has lower loss than continued editing ($+0.00442$ shared, $+0.00360$ separate); in both regimes its 90% interval is not inside $\pm\delta$. Pre-declared expectations E1 (continued below frozen in both factors), E4 (at least 90% of checkpoints `history/single` in both factors) and E6 (small factor differences) fail for the separate factor or for P1; E2, E3 and E5 (except P2 under shared history) hold.

Between-lineage variance and lock-in. Under shared history the four lineages of a family reach the same checkpoint behavior in 126/126 families; under separate histories they do so in 31/126. The between-lineage share of variance of continued minus frozen is 0.24 (shared) and 0.47 (separate); for P2 under shared history it is 0.64, the one primary contrast in which source evidence varies

more than the world. Under shared history, lineage-0-only estimates agree with the four-lineage estimates in sign and detection for P1 and P3, while P2 is not detected with one lineage; under separate histories the lineage-0-only P3 is not detected.

Mechanism. The verification traced the lock-in to the length of the source history, not to the content of the fixed-meta records. The base program's successor generator (meta rule `cycle`) starts its proposal window at `len(history)` mod 25, 25 being the number of legal single edits of the base program, so its proposals depend only on how many records the history holds. Both factors begin with two records at stage 0. At later stages the inherited history holds 13, 24 and 35 records under shared history and 9, 16 and 23 under separate histories, because the fixed-meta candidate records add four per stage. Under shared history the stage-1 window offers only single-term variants and the path ends at `history/single`; under separate histories the windows offer paired variants and the search usually ends at `residual/paired`. Replacing the fixed-meta records in the separate-history run by the same number of content-free placeholder records reproduces the shared-history checkpoint program in 504/504 lineages. Continued through E3, the placeholder run gives `history/single` checkpoints in 504/504 lineages, frozen transfer losses identical to shared history (maximum absolute difference 0), identical continued transfer in 486/504 runs, and P1 -0.00890 ($[-0.01288, -0.00525]$), P2 -0.00451 and P3 $+0.00455$ (Table 20, verification rows), essentially the shared-history values. The contents of the fixed-meta records, which the credit and compose meta rules can read, are therefore irrelevant to the checkpoint and nearly irrelevant to E3.

The single-term lock-in, and with it most of the continued-over-frozen gain of every earlier cohort, thus depends on an incidental feature of the shared-history design: the history length sets which edits the base generator proposes. It is not a robust consequence of inherited editing. Neither factor is an unbiased reference: the separate-history search is governed by the same length-indexed rotation and reaches a different path only because its history is shorter. We therefore read the pair of factors as showing that P1 (continued minus frozen) is fragile under a change of history contract, while P3 (validation selection among fixed behaviors has lower loss than continued editing) holds under both. Consistent with this reading, conditional on the separate-history checkpoint behavior, the continued-over-frozen gain appears only in lineages whose checkpoint uses single-term repair.

E.5 Lifetime cost

This analysis is declared in the lineage protocol as post-hoc and descriptive; it is not part of the primary family. Its assumptions are stored verbatim in the lineage summary file. (i) The unit of cost is one ridge fit, the unit of every budget in the study. (ii) A deployment is T future transfer tasks drawn like the frozen transfer tasks; a method's per-task loss in a family is its mean over the four evaluated transfer tasks and does not change with T (no learning during deployment). (iii) Source training is a sunk cost shared by all methods. (iv) Continued editing, fixed meta and frozen-with-continuation pay the measured continuation cost of 192 fits once (four stages of 48 credits), then 3 fits per task. (v) The frozen checkpoint without continuation pays 0 up front and 3 fits per task. (vi) Validation selection among the six behaviors pays 0 up front and 18 nominal fits per task (5.65 distinct fits per task if fits are cached). (vii) At equal budget $B(T) = 192 + 3T$, validation selection can afford $k(T) = \min(6, 1 + 64/T)$ behaviors per task, drawn uniformly at random with exact averaging over subsets and linear interpolation for fractional k.

Results. Full validation selection uses fewer cumulative fits than continued editing for $T \leq 12$ ($192/15 = 12.8$), and over this range it also has lower loss (main replication $+0.00262$, $[+0.00026, +0.00528]$). At equal budget (Table 22, Figure 10) the mean difference changes sign at $T = 51$ under shared history, and continued editing is detectably better from $T = 80$ (at $T = 500$: -0.00494). Under separate histories the corresponding horizons are $T = 44$ and $T = 89$. As $k \to 1$ the budget-matched selector becomes a uniformly random behavior, which continued editing beats in the saved cohorts (main replication -0.00593, $[-0.00866, -0.00345]$); the large-T advantage therefore reflects mainly the selector's shrinking per-task budget. Under separate histories the zero-continuation frozen checkpoint has loss 0.8726 against 0.8714 for continued editing, so the 192 continuation fits buy no detectable gain at any horizon.

Table 20: Independent source lineages (126 families × 4 lineages). Family value = mean over its lineages; bootstrap over families. Holm over the six primary tests (P1–P3 × 2 factors). B/T/W = families with continued better/tied/worse. s^2_{lin} = between-lineage share of the variance of the 504 lineage values.

Contrast	Mean	95% CI	90% CI	Holm p	B/T/W	s^2_{lin}	Verdict
Shared source history (all earlier cohorts)							
P1 continued−frozen	−0.00902	[−0.01292, −0.00543]	[−0.01228, −0.00599]	$< 10^{-4}$	102/2/22	0.24	det., relevant
P2 continued−fixed meta	−0.00302	[−0.00550, −0.00079]	[−0.00504, −0.00111]	0.020	60/32/34	0.64	det., not equiv.
P3 continued−val-selected (source history)	+0.00442	[+0.00157, +0.00790]	[+0.00193, +0.00726]	0.004	31/18/77	0.29	det., not equiv.
continued−val-selected (continued history)	+0.00409	[+0.00158, +0.00700]	[+0.00192, +0.00645]	—	29/18/79	0.41	det., not equiv.
continued−oracle behavior	+0.00731	[+0.00432, +0.01094]	[+0.00472, +0.01031]	—	1/5/120	0.28	det., not equiv.
continued−uniform random behavior	−0.00556	[−0.00782, −0.00339]	[−0.00744, −0.00376]	—	105/0/21	0.47	det., not equiv.
continued−fixed-meta source lineage	−0.00095	[−0.00293, +0.00077]	[−0.00256, +0.00051]	—	48/45/33	0.74	equivalent
P1, lineage 0 only	−0.00910	[−0.01367, −0.00498]	[−0.01283, −0.00563]	—	96/12/18	—	det., relevant
P2, lineage 0 only	−0.00354	[−0.00845, +0.00083]	[−0.00750, +0.00020]	—	44/68/14	—	not detected
P3, lineage 0 only	+0.00455	[+0.00073, +0.00906]	[+0.00125, +0.00824]	—	32/22/72	—	det., not equiv.
Separate source histories							
P1 continued−frozen	−0.00121	[−0.00316, +0.00121]	[−0.00289, +0.00077]	0.56	54/52/20	0.47	equivalent
P2 continued−fixed meta	−0.00098	[−0.00288, +0.00141]	[−0.00258, +0.00096]	0.56	43/65/18	0.43	equivalent
P3 continued−val-selected (source history)	+0.00360	[+0.00079, +0.00711]	[+0.00116, +0.00645]	0.018	32/27/67	0.15	det., not equiv.
continued−val-selected (continued history)	+0.00364	[+0.00082, +0.00713]	[+0.00120, +0.00649]	—	30/27/69	0.15	det., not equiv.
continued−oracle behavior	+0.00683	[+0.00333, +0.01113]	[+0.00380, +0.01036]	—	1/8/117	0.02	det., not equiv.
continued−uniform random behavior	−0.00614	[−0.00899, −0.00333]	[−0.00855, −0.00379]	—	108/0/18	0.03	det., not equiv.
continued−fixed-meta source lineage	−0.00192	[−0.00440, +0.00079]	[−0.00398, +0.00032]	—	64/36/26	0.39	equivalent
P1, lineage 0 only	−0.00148	[−0.00439, +0.00184]	[−0.00393, +0.00124]	—	29/89/8	—	equivalent
P2, lineage 0 only	−0.00012	[−0.00247, +0.00293]	[−0.00211, +0.00239]	—	20/96/10	—	equivalent
P3, lineage 0 only	+0.00231	[−0.00013, +0.00550]	[+0.00017, +0.00491]	—	32/29/65	—	equivalent
Verification: shared history length with content-free placeholders (no fixed-meta records)							
P1 continued−frozen	−0.00890	[−0.01288, −0.00525]	[−0.01221, −0.00579]	—	101/2/23	—	det., relevant
P2 continued−fixed meta	−0.00451	[−0.00787, −0.00156]	[−0.00724, −0.00199]	—	63/31/32	—	det., not equiv.
P3 continued−val-selected (source history)	+0.00455	[+0.00162, +0.00812]	[+0.00201, +0.00747]	—	31/18/77	—	det., not equiv.
Separate − shared (paired within family and lineage)							
P1 continued−frozen	+0.00782	[+0.00404, +0.01258]	[+0.00455, +0.01165]	—	24/4/98	0.38	det., not equiv.
P2 continued−fixed meta	+0.00204	[−0.00124, +0.00609]	[−0.00076, +0.00535]	—	51/22/53	0.51	not detected
P3 continued−val-selected (source history)	−0.00083	[−0.00295, +0.00168]	[−0.00265, +0.00122]	—	44/55/27	0.57	equivalent
continued−fixed-meta source lineage	−0.00096	[−0.00396, +0.00291]	[−0.00353, +0.00222]	—	61/25/40	0.56	equivalent
arm loss: continued	−0.00049	[−0.00239, +0.00192]	[−0.00215, +0.00147]	—	45/54/27	0.50	equivalent
arm loss: frozen	−0.00831	[−0.01178, −0.00508]	[−0.01122, −0.00559]	—	110/0/16	0.27	det., relevant
arm loss: fixed meta	−0.00254	[−0.00541, +0.00009]	[−0.00495, −0.00031]	—	60/31/35	0.49	equivalent

Verdict: *det.* = 95% percentile interval excludes 0; *equiv.* = 90% interval inside $\pm\delta$; *not equiv.* = detected, 90% interval not inside $\pm\delta$; *relevant* = detected, 90% interval entirely outside $\pm\delta$; *not detected* = neither.

Table 21: Independent lineages: checkpoint behaviors (lineages out of 504 per factor), lock-in and realized fits.

	Shared	Separate
Checkpoint history/single	504	75
Checkpoint residual/paired	0	365
Checkpoint history/paired	0	50
Checkpoint residual/single	0	14
Families whose 4 lineages share one checkpoint behavior	126/126	31/126
Distinct checkpoint programs per family (mean)	1.76	2.10
Continued final: paired repair	465	473
Source fixed meta: paired repair	336	314
Source fits (both source lineages)	252	264
E3 continuation fits per arm	192	192

Table 22: Lifetime cost at equal cumulative fit budget $B(T) = 192 + 3T$ over T future transfer tasks. $k(T)$ = behaviors per task that budget-matched validation selection can afford. Entries: continued minus budget-matched selection (negative = continued better), hierarchical bootstrap over 126 families.

		Shared source history		Separate source histories	
T	$k(T)$	Mean	95% CI	Mean	95% CI
1	6.00	+0.00442	[+0.00157, +0.00790]	+0.00360	[+0.00079, +0.00711]
2	6.00	+0.00442	[+0.00157, +0.00790]	+0.00360	[+0.00079, +0.00711]
5	6.00	+0.00442	[+0.00157, +0.00790]	+0.00360	[+0.00079, +0.00711]
10	6.00	+0.00442	[+0.00157, +0.00790]	+0.00360	[+0.00079, +0.00711]
12	6.00	+0.00442	[+0.00157, +0.00790]	+0.00360	[+0.00079, +0.00711]
13	5.92	+0.00439	[+0.00156, +0.00784]	+0.00358	[+0.00078, +0.00707]
20	4.20	+0.00341	[+0.00113, +0.00608]	+0.00277	[+0.00034, +0.00597]
32	3.00	+0.00182	[0.00000, +0.00391]	+0.00130	[−0.00085, +0.00424]
50	2.28	+0.00002	[−0.00154, +0.00175]	−0.00048	[−0.00252, +0.00220]
64	2.00	−0.00068	[−0.00219, +0.00096]	−0.00117	[−0.00323, +0.00143]
100	1.64	−0.00244	[−0.00407, −0.00073]	−0.00296	[−0.00521, −0.00042]
128	1.50	−0.00312	[−0.00485, −0.00134]	−0.00366	[−0.00602, −0.00108]
200	1.32	−0.00400	[−0.00588, −0.00211]	−0.00455	[−0.00706, −0.00193]
300	1.21	−0.00452	[−0.00651, −0.00254]	−0.00508	[−0.00771, −0.00242]
500	1.13	−0.00494	[−0.00704, −0.00287]	−0.00550	[−0.00822, −0.00278]
first T, mean < 0		51		44	
first T, 95% CI < 0		80		89	
same, lineage 0 only		53 / 118		39 / 81	

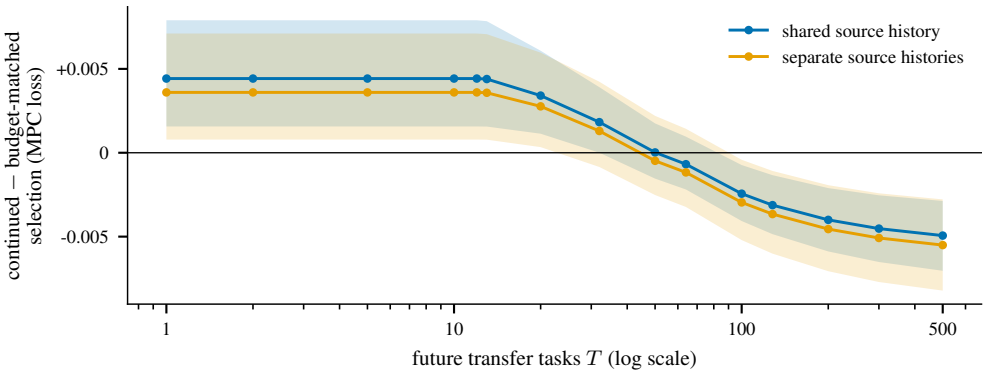

Figure 10: Continued editing minus budget-matched validation selection as a function of the number of future transfer tasks T (values of Table 22; bands: 95% hierarchical bootstrap intervals). Below zero, continued editing has lower loss at equal cumulative fit budget.

E.6 STATISTICAL RE-ANALYSIS

Methods. All numbers in this subsection come from saved per-family outcomes, except the compute-matched baseline, which re-executes frozen code. Percentile intervals reuse the resample matrix of the original analysis; BCa intervals use the same resamples with jackknife acceleration. The Wilcoxon signed-rank test (two-sided) and the exact sign test use non-zero differences ($|d| > 10^{-12}$). Holm adjustment is applied separately to Wilcoxon and sign p-values within each cohort's declared primary family: the four main-protocol primaries for the original main cohort, the main replication and (declared in the analysis script, since its protocol lists none) the misspecification cohort; the follow-up primary and its two replication contrasts for both follow-up cohorts; P2 and P3 for the curriculum test. Post-hoc contrasts form a separate Holm family per cohort. The δ-decision is repeated at $\delta \in \{0.0025, 0.005, 0.01\}$.

Holm per cohort. In the original main cohort only continued minus frozen survives Holm (1 of 4); E1, E2 and continued minus fixed meta do not, although their BCa intervals exclude zero. In the main replication all four primaries survive (4 of 4) (Table 23). The replication is therefore the primary cohort for every claim in the paper; the original cohort is reported as a discovery cohort.

δ-sensitivity. Continued minus frozen lies beyond the margin in the original and main-replication cohorts but not in the follow-up replication, misspecification or curriculum cohorts, where it is detected but its 90% interval reaches inside the margin. The generator lock is equivalent at every δ, including 0.0025 (smallest δ giving equivalence 0.00219). Continued minus validation-selected in the main replication ($+0.00262$, 90% CI $[+0.00060, +0.00480]$) is detected and practically equivalent at $\delta = 0.005$ and 0.01, but not at 0.0025 (smallest equivalence δ: 0.00480); it supports "detectably but not practically different", not "outperforms" (Table 24).

Influence. In the original cohort one family contributes 77% of the summed E1 procedure effect and 51% of the E2 interaction; without it neither is detected (Table 25). In the replication the E1 procedure effect survives removal of the top three families at about a quarter of its size, and the E2 interaction survives removal of the top family. Per-family differences are heavy-tailed throughout: medians are zero or below 10^{-3} in magnitude. Procedure shares should be read with their width: 38% (95% bootstrap [3%, 46%]) in the original cohort, 45% ([30%, 50%]) in the replication and 38% ([22%, 63%]) under misspecification. The near-identical original and misspecification shares were checked against the raw outcome files: the cohorts share no seeds, outcome files or non-zero per-family values, so the agreement is a numerical coincidence.

Compute-matched non-recursive baseline. For each family, one of the six behaviors is chosen using only the lineage's own continuation evidence: each behavior repairs each of the eight E3 continuation tasks once from the empty model with 3 fits, starting from the checkpoint history plus the continuation evidence and appending its own records, and the behavior with the lowest mean validation error is selected. It is then evaluated on the four transfer tasks with 3 fits per task, like the lineage. Selection costs exactly 144 fits per family, against the lineage's 192 continuation credits. The frozen E3 code is re-executed and the continued arm reproduces its saved transfer losses exactly in every family. The baseline selects `residual/paired` in 229/252 replication families (26/32 original) and never selects single-term repair. Continued minus matched baseline is $+0.00142$ (95% CI $[-0.00059, +0.00370]$, 90% CI $[-0.00029, +0.00329]$; better/tie/worse 29/184/39) in the replication and $+0.00136$ ($[-0.00026, +0.00408]$) in the original cohort: practically equivalent at $\delta = 0.005$ in both, with the point estimate favouring the baseline. With the same evidence and fewer fits, choosing one fixed behavior per family matches the self-editing lineage within the margin.

Power. The cohort size of 252 was chosen for rotation balance, not by a power calculation (Table 26). Evaluated post hoc at the $n = 32$ standard deviations, 47 families would give a 90% half-width below $\delta/2$ for the generator-lock contrast, and 252 gives TOST power above 0.999 at a true difference of zero; the achieved half-width is 0.0020. For continued minus fixed meta in the main protocol, 302 families would be needed and 252 gives TOST power 0.83. The replication is thus adequately powered, post hoc, for the generator-lock equivalence; we do not claim it is powered for every contrast.

Table 23: Round-3 re-analysis of the declared primary contrasts; Holm adjustment within each cohort's declared family (misspecification: the main family, declared in the analysis script). Negative = first term better.

Contrast	Mean	95% pct. CI	95% BCa CI	90% pct. CI	W p	sign p	Holm W / sign p	δ: .0025/.005/.01
Original main ($n = 32$, seeds 8101–8132)								
E1 procedure main effect	-0.00101	$[-0.00271, -0.00002]$	$[-0.00455, -0.00014]$	$[-0.00253, -0.00007]$	0.077	0.15	0.15 / 0.44	R/DE/DE
E2 own−replay interaction	$+0.00257$	$[+0.00003, +0.00622]$	$[+0.00019, +0.00842]$	$[+0.00013, +0.00533]$	0.12	0.38	0.15 / 0.44	R/R/DE
continued−frozen	-0.01414	$[-0.02527, -0.00462]$	$[-0.02693, -0.00580]$	$[-0.02331, -0.00598]$	3×10^{-5}	8×10^{-6}	1×10^{-4} / 3×10^{-5}	R†/R†/R
continued−fixed meta	-0.00988	$[-0.01961, -0.00208]$	$[-0.02165, -0.00282]$	$[-0.01798, -0.00281]$	0.021	0.21	0.064 / 0.44	R†/R/R
Original follow-up ($n = 32$, seeds 9401–9432)								
continued−generator-locked	-0.00086	$[-0.00491, +0.00232]$	$[-0.00656, +0.00156]$	$[-0.00409, +0.00160]$	0.74	1.00	0.74 / 1.00	N/E/E
continued−frozen	-0.01080	$[-0.02201, -0.00264]$	$[-0.02801, -0.00409]$	$[-0.01992, -0.00342]$	4×10^{-4}	2×10^{-4}	0.001 / 6×10^{-4}	R†/R/R
continued−fixed meta	-0.00013	$[-0.00192, +0.00199]$	$[-0.00176, +0.00239]$	$[-0.00171, +0.00157]$	0.23	0.11	0.46 / 0.22	E/E/E
Main replication ($n = 252$, seeds 8201–8452)								
E1 procedure main effect	-0.00406	$[-0.00936, -0.00087]$	$[-0.01374, -0.00138]$	$[-0.00837, -0.00109]$	8×10^{-11}	2×10^{-7}	2×10^{-10} / 5×10^{-7}	R/R/DE
E2 own−replay interaction	$+0.00987$	$[+0.00036, +0.02570]$	$[+0.00202, +0.04026]$	$[+0.00142, +0.02261]$	0.006	0.009	0.006 / 0.016	R/R/R
continued−frozen	-0.01010	$[-0.01501, -0.00579]$	$[-0.01566, -0.00620]$	$[-0.01412, -0.00653]$	5×10^{-21}	2×10^{-30}	2×10^{-20} / 8×10^{-30}	R†/R†/R
continued−fixed meta	-0.00129	$[-0.00274, +0.00002]$	$[-0.00288, -0.00006]$	$[-0.00248, -0.00019]$	0.002	0.008	0.004 / 0.016	E/E/E
Follow-up replication ($n = 252$, seeds 9501–9752)								
continued−generator-locked	$+0.00031$	$[-0.00217, +0.00251]$	$[-0.00242, +0.00234]$	$[-0.00171, +0.00219]$	0.15	0.097	0.15 / 0.097	E/E/E
continued−fixed meta	-0.00425	$[-0.00704, -0.00187]$	$[-0.00766, -0.00221]$	$[-0.00655, -0.00221]$	3×10^{-5}	2×10^{-4}	6×10^{-5} / 4×10^{-4}	R/R/DE
continued−frozen	-0.00697	$[-0.01024, -0.00397]$	$[-0.01078, -0.00426]$	$[-0.00969, -0.00440]$	6×10^{-18}	4×10^{-29}	2×10^{-17} / 1×10^{-28}	R†/R/DE
Misspecification ($n = 252$, seeds 8601–8852)								
E1 procedure main effect	-0.00491	$[-0.00858, -0.00198]$	$[-0.01024, -0.00250]$	$[-0.00793, -0.00235]$	3×10^{-7}	4×10^{-5}	9×10^{-7} / 1×10^{-4}	R/R/DE
E2 own−replay interaction	$+0.01672$	$[+0.00568, +0.03002]$	$[+0.00685, +0.03292]$	$[+0.00703, +0.02751]$	0.066	0.23	0.13 / 0.46	R†/R†/R
continued−frozen	-0.00757	$[-0.01264, -0.00237]$	$[-0.01254, -0.00226]$	$[-0.01178, -0.00318]$	2×10^{-10}	4×10^{-12}	9×10^{-10} / 2×10^{-11}	R†/R/R
continued−fixed meta	$+0.00128$	$[-0.00261, +0.00577]$	$[-0.00213, +0.00653]$	$[-0.00200, +0.00501]$	0.24	0.27	0.24 / 0.46	N/N/E
Curriculum ($n = 252$, seeds 8901–9152)								
P2 DiD	-0.00029	$[-0.00114, +0.00036]$	$[-0.00145, +0.00025]$	$[-0.00096, +0.00028]$	0.37	1.00	0.73 / 1.00	E/E/E
P3 frozen MIXED−ORIGINAL	-0.00007	$[-0.00022, 0.00000]$	$[-0.00038, 0.00000]$	$[-0.00022, 0.00000]$	1.00	1.00	1.00 / 1.00	E/E/E

Decisions at each δ: R = detected, not shown within $\pm\delta$ (R†: 90% CI beyond the margin at $\delta = 0.005$); DE = detected but practically equivalent; E = equivalent (not detected); N = not detected, not equivalent. Wilcoxon signed-rank and exact sign tests are two-sided on non-zero differences.

Table 24: Round-3 re-analysis of post-hoc contrasts (separate Holm family per cohort; undeclared analyses).

Contrast	Mean	95% pct. CI	95% BCa CI	90% pct. CI	W p	sign p	Holm W / sign p	δ: .0025/.005/.01
Original main ($n = 32$, seeds 8101–8132)								
E1 naive descendant gain	-0.00266	$[-0.00605, -0.00066]$	$[-0.00915, -0.00093]$	$[-0.00565, -0.00078]$	5×10^{-5}	9×10^{-5}	3×10^{-4} / 4×10^{-4}	R/R/DE
crossing: code effect	-0.01414	$[-0.02526, -0.00461]$	$[-0.02693, -0.00580]$	$[-0.02331, -0.00598]$	3×10^{-5}	8×10^{-6}	2×10^{-4} / 5×10^{-5}	R†/R†/R
crossing: history effect	0.00000	$[-0.00001, 0.00000]$	$[-0.00001, 0.00000]$	$[-0.00001, 0.00000]$	1.00	1.00	1.00 / 1.00	E/E/E
crossing: code×history	0.00000	$[-0.00001, 0.00000]$	$[-0.00002, 0.00000]$	$[-0.00001, 0.00000]$	1.00	1.00	1.00 / 1.00	E/E/E
continued−val-selected (per task)	$+0.00411$	$[-0.00009, +0.01001]$	$[+0.00066, +0.01265]$	$[+0.00050, +0.00891]$	0.021	0.064	0.085 / 0.26	N/N/E
continued−compute-matched baseline	$+0.00136$	$[-0.00026, +0.00408]$	$[-0.00008, +0.00649]$	$[-0.00013, +0.00363]$	0.21	0.23	0.62 / 0.68	N/E/E
same, checkpoint history	$+0.00136$	$[-0.00026, +0.00408]$	$[-0.00008, +0.00649]$	$[-0.00013, +0.00363]$	0.21	0.23	0.82 / 1.00	N/E/E
Main replication ($n = 252$, seeds 8201–8452)								
E1 naive descendant gain	-0.00895	$[-0.01962, -0.00248]$	$[-0.02846, -0.00358]$	$[-0.01759, -0.00295]$	4×10^{-29}	6×10^{-24}	2×10^{-28} / 3×10^{-23}	R†/R/R
crossing: code effect	-0.01029	$[-0.01510, -0.00615]$	$[-0.01591, -0.00657]$	$[-0.01426, -0.00679]$	5×10^{-21}	2×10^{-30}	2×10^{-20} / 1×10^{-29}	R†/R†/R
crossing: history effect	$+0.00019$	$[-0.00002, +0.00060]$	$[-0.00002, +0.00099]$	$[-0.00002, +0.00059]$	0.62	0.38	0.62 / 0.82	E/E/E
crossing: code×history	-0.00042	$[-0.00124, 0.00000]$	$[-0.00241, 0.00000]$	$[-0.00122, 0.00000]$	0.31	0.38	0.62 / 0.82	E/E/E
continued−val-selected (per task)	$+0.00262$	$[+0.00026, +0.00528]$	$[+0.00055, +0.00577]$	$[+0.00060, +0.00480]$	4×10^{-5}	6×10^{-4}	2×10^{-4} / 0.002	R/DE/DE
continued−compute-matched baseline	$+0.00142$	$[-0.00059, +0.00370]$	$[-0.00036, +0.00409]$	$[-0.00029, +0.00329]$	0.18	0.27	0.53 / 0.82	N/E/E
same, checkpoint history	$+0.00078$	$[-0.00093, +0.00257]$	$[-0.00085, +0.00264]$	$[-0.00064, +0.00223]$	0.20	0.27	0.71 / 1.00	E/E/E
Misspecification ($n = 252$, seeds 8601–8852)								
E1 naive descendant gain	-0.01291	$[-0.02152, -0.00515]$	$[-0.02297, -0.00607]$	$[-0.01991, -0.00639]$	1×10^{-16}	1×10^{-13}	1×10^{-16} / 1×10^{-13}	R†/R†/R

Decisions at each δ: R = detected, not shown within $\pm\delta$ (R†: 90% CI beyond the margin at $\delta = 0.005$); DE = detected but practically equivalent; E = equivalent (not detected); N = not detected, not equivalent. Wilcoxon signed-rank and exact sign tests are two-sided on non-zero differences.

Table 25: Influence of the most extreme families (by $|d|$) on the E1 procedure effect and the E2 interaction. Share = contribution of the top family (top three families) to the sum of per-family differences.

		share of sum		excluding top 1			excluding top 3		
Cohort		top 1	top 3	mean	95% CI	W p	mean	95% CI	W p
Orig.	E1	77%	96%	-0.00024	$[-0.00055, +0.00003]$	0.15	-0.00005	$[-0.00021, +0.00012]$	0.50
Orig.	E2	51%	99%	$+0.00130$	$[0.00000, +0.00370]$	0.25	$+0.00003$	$[0.00000, +0.00009]$	1.00
Rep.	E1	53%	76%	-0.00190	$[-0.00361, -0.00060]$	1×10^{-10}	-0.00100	$[-0.00185, -0.00032]$	4×10^{-10}
Rep.	E2	65%	96%	$+0.00347$	$[+0.00003, +0.00851]$	0.010	$+0.00036$	$[-0.00001, +0.00106]$	0.026
Missp.	E1	27%	47%	-0.00361	$[-0.00594, -0.00157]$	5×10^{-7}	-0.00263	$[-0.00450, -0.00102]$	1×10^{-6}
Missp.	E2	18%	51%	$+0.01377$	$[+0.00400, +0.02557]$	0.088	$+0.00831$	$[+0.00110, +0.01681]$	0.15

Table 26: Post-hoc power for the equivalence tests ($\delta = 0.005$; normal approximation, 90% half-width $= 1.645\,\mathrm{SD}/\sqrt{n}$). "$n$ needed": half-width below $\delta/2$ at the $n = 32$ SD.

Contrast	SD ($n = 32$)	n needed	pred. half-width ($n = 252$)	TOST power ($n = 252$, true 0)	SD ($n = 252$)	obs. half-width ($n = 252$)
continued−generator-locked	0.01037	47	0.00107	> 0.999	0.01883	0.00195
continued−fixed meta (follow-up)	0.00577	15	0.00060	> 0.999	0.02108	0.00217
continued−fixed meta (main)	0.02639	302	0.00273	0.83	0.01109	0.00115

F Interventions, replay and the unit of evidence

This appendix makes the scope of the comparisons explicit. The identities and counterexamples are elementary evaluation arguments, not new statistical guarantees. They specify which claims the implemented tests can support and which require further evidence.

F.1 The checkpoint is part of the intervention

Let $X = (M, P, H, L, E, \omega; K)$ be the checkpoint in the main text. Replacing P is a well-defined intervention only when its interfaces and derived state remain compatible. Stale residuals, changed normalization, hidden archives or unrecorded simulator randomness can create unintended simultaneous interventions. Recomputing a cache from common raw evidence is permissible when its cost is charged.

A sham restoration leaves scientific contents unchanged and should reproduce candidate identities, actions, observations, predictions and utility within the declared tolerance. Restoring an ancestor under current permitted evidence checks a different intervention; it need not recover the ancestor's historical score. Neither check proves compatibility with every future edit. Table 27 maps mismatches to observable checks.

State component	Hidden mismatch	Observable check
Model	Changed normalization or stale residual cache.	Prediction equality and cache reconstruction.
Procedure	Different interpreter or inactive edits.	Same-input behavior and executed-path witness.
Evidence	Future labels or failed trials omitted.	Access log and identical permitted prefixes.
Environment	Shared mutable state across branches.	Fresh branch restoration and action-conditioned outcomes.
Randomness	Action-dependent stream misalignment.	Declared noise coupling and stream ownership.
Resources	Rejected edits charged to only one arm.	Reconstructed actual call ledger.

Table 27: Checkpoint failure modes and corresponding evidence. These checks concern attribution, not general safety.

F.2 Four cells and two conventions for main effects

For losses Y_{ij}, define ancestor-conditional effects $d_P = Y_{01} - Y_{00}$ and $d_M = Y_{10} - Y_{00}$, and interaction $I = Y_{11} - Y_{10} - Y_{01} + Y_{00}$. Then

$$Y_{11} - Y_{00} = d_P + d_M + I. \tag{6}$$

The factorial main effects average across the other factor:

$$\Delta_P = \tfrac{1}{2}[(Y_{01} - Y_{00}) + (Y_{11} - Y_{10})] = d_P + I/2, \tag{7}$$

$$\Delta_M = \tfrac{1}{2}[(Y_{10} - Y_{00}) + (Y_{11} - Y_{01})] = d_M + I/2. \tag{8}$$

Thus $Y_{11} - Y_{00} = \Delta_P + \Delta_M$. Adding I again to these averaged effects would double count it. The main experiment declares which convention it uses. Neither convention identifies the historical causal process that learned the two checkpoints; it identifies the effects of executing the crossed versions in the specified evaluation.

A diagonal improvement cannot identify the procedure effect. Table 28 gives two hypothetical schedules with identical ancestor and descendant losses: In schedule B, the model improvement

Schedule	Y_{00}	Y_{01}	Y_{10}	Y_{11}
A	4	2	4	2
B	4	5	1	2

Table 28: Illustrative values, not experimental measurements. Both diagonal comparisons improve from four to two. In A the procedure main effect is -2; in B it is $+1$, despite the same descendant gain.

masks a harmful procedure change. Supplying the crossed cells resolves this ambiguity for the tested states. It does not ensure that a transported procedure will work with an unrelated model architecture.

F.3 Why an adaptive branch needs its own observations

Consider actions a and b in a one-step environment. A donor always takes a and records reward zero. Two environments agree on that entire log but assign b rewards $+1$ and -1. A target always choosing b has opposite values. Therefore no function of the donor log alone identifies its adaptive value.

Additional evidence or assumptions can resolve the ambiguity: an executable simulator, a logging policy with action support, or an appropriate estimator with known propensities and valid models. Exact-tuple replay remains useful for comparing procedures on supplied evidence. It becomes misleading only when that conditional comparison is presented as the value of unexecuted target actions.

Our adaptive branches execute their probes; replay branches consume an explicit donor's unchanged tuples. Their contrast measures sensitivity to evidence regime. No natural direct/indirect-effect decomposition is identified because the required mediation assumptions are not supplied.

F.4 Paired estimates and uncertainty

For independent world families $i = 1, \ldots, n$, let d_i be the paired contrast after averaging that family's declared tasks. The estimator is $\bar{d} = n^{-1} \sum_i d_i$. Bootstrap resampling preserves the complete paired family. Shared exogenous draws can reduce variance without replacing action-conditioned execution; correlated transitions do not increase n.

These empirical intervals describe variation within the programmed distribution, without finite-sample coverage or external-validity guarantees. They do not account for choosing a new analysis after viewing outcomes. Such analyses need explicit exploratory status. Target tasks from one frozen repairer estimate its utility; independently learned source lineages support a broader training-process comparison. Those levels of replication must remain distinct.

F.5 Fixed code is not the same as fixed behavior

Fixed update code can use memory, fit a utility predictor and adapt its search distribution. Mutable code can be behaviorally redundant. The relevant comparison therefore specifies information, reachable functions and budgets, rather than assigning capability levels to the labels fixed and recursive.

A bounded editable interpreter can also be represented as a fixed transition rule over a state containing its program. This observation preserves meaningful differences in edit permissions, search cost and inductive bias; it prevents the word recursive alone from establishing a capability advantage.

F.6 Retention, costs and selection of descendants

Retention uses protected prior-regime cases; a repeatedly queried canary is development feedback. Passing checked cases does not prove universal preservation. Any acceptable tradeoff between retention and new-task utility must be declared before selecting an edit.

Lifetime cost includes development and deployment over an explicit horizon. Charge source observations, rejected candidates and repeated computation; report unspent allocations separately. A frozen procedure may spend saved proposal credits on ordinary repair when the contract permits, although that need not create useful new search.

Selecting the best descendant using final outcomes estimates a retrospective best-of-lineage policy, not the prespecified training process. The latter selects with development feedback and evaluates the resulting process on protected outcomes. Rejections, empty updates and adverse continuations remain part of its accounting.

G Prior work and the scope of procedural attribution

This comparison asks which object changes: a prediction, a world model, a task-solving policy, a repair procedure, or the procedure that produces subsequent procedures. These objects can improve together, so a rising task score does not identify which change caused the gain. Our study concerns inherited diagnostic and repair state acting on an executable transition model. It does not claim that editable agents, learned diagnosis, model repair, or controlled evaluation originate here. Tables 29 and 30 map model antecedents and evaluation controls.

World models and the learned transition component. World-model research already separates learned environment prediction from behavior generated using that prediction. World Models combines learned representations, dynamics, and control (Ha & Schmidhuber, 2018); PlaNet plans through learned latent dynamics (Hafner et al., 2019). DreamerV2 and DreamerV3 learn behavior using imagined outcomes (Hafner et al., 2021; 2024), while TD-MPC2 performs trajectory optimization in an implicit latent model (Hansen et al., 2024). These methods motivate evaluating both prediction and downstream decision quality, but they are not reproduced baselines for this small executable-model setting. Their representation learning and scale should not be attributed to our implementation.

The implemented neural component uses fixed, explicitly specified nonlinear features and learned output coefficients. Extreme learning machines provide a general precedent for fitting output coefficients over fixed hidden features (Huang et al., 2006); the present features are specified rather than randomly sampled, and this is not end-to-end representation learning or a random Fourier feature construction. Sparse identification of nonlinear dynamics selects parsimonious equations from a candidate function library (Brunton et al., 2016). This supplies a relevant system-identification comparison: identifying a useful dynamics model is already a learning problem, whereas our additional question is whether inherited repair state changes the efficiency or reliability of that identification. Fitting new coefficients, selecting symbolic terms, and changing the procedure that selects or fits them remain distinct interventions.

Executable and neuro-symbolic model revision. Learning formal action models from observations predates current language-model systems. ARMS uses weighted satisfiability to infer action descriptions from successful plans with incomplete intermediate states (Yang et al., 2007). WorldCoder synthesizes executable transition and reward programs and refines them using interaction

Table 29: Selected world-model antecedents and their relation to the present repairer-attribution question.

Work	Documented mutable object	Relevant difference
WorldCoder	Executable transition/reward programs	Model refinement versus inherited repair state
WorldCloner	Symbolic rules and adapted policy	World-model change versus updater change
VisualPredicator	Predicates and abstract action model	Representation invention versus repairer attribution
CoEx	Beliefs and exploration context	Coupled interaction versus controlled state transfer
PoE-World	Programmatic expert composition	Model composition versus procedure inheritance
DreamCoder	Library and recognition guidance	Learned search guidance versus an edited update procedure

evidence (Tang et al., 2024). WorldCloner responds to prediction violations through symbolic-rule revision and neural-policy adaptation (Balloch et al., 2023). Thus, neither executable dynamics nor failure-triggered world-model repair is sufficient novelty. VisualPredicator learns predicates and abstract action models, including proposals informed by successful and failed transitions (Liang et al., 2025). CoEx couples exploration and belief updates (Kim & won Hwang, 2025); choosing informative interactions is consequently another established ingredient.

Compositional representations have equally direct antecedents. PoE-World combines programmatic experts into a world model (Piriyakulkij et al., 2025), and DreamCoder jointly develops reusable symbolic abstractions and neural guidance for program search (Ellis et al., 2020). Conversational proprioception augments neuro-symbolic world models with action memory and constraints (Agravante et al., 2023); Cosmos binds neural and symbolic entity descriptions to interaction rules (Sehgal et al., 2024). These works inform possible representation extensions, but the present finite edit space does not establish open-ended predicate invention or reusable library discovery.

The phrase neuro-symbolic also covers materially different couplings. DeepProbLog combines neural predicates with probabilistic logic (Manhaeve et al., 2018); Logic Tensor Networks use differentiable many-valued first-order logic (Badreddine et al., 2022). The Neuro-Symbolic Concept Learner connects visual concepts with executable symbolic programs (Mao et al., 2019), and semantic loss constrains neural outputs using logical knowledge (Xu et al., 2018). Neural DNF-MT additionally permits learned policy representations to be translated into editable logic programs (Baugh et al., 2025). Our symbolic forces coupled to a context-conditioned neural action term require their own operational definition; they inherit none of these systems' logical guarantees merely from sharing a label.

Learning an update rule and editing an improver. Meta-learning already optimizes how a system adapts. MAML learns an initialization for subsequent adaptation (Finn et al., 2017); Reptile studies a first-order alternative (Nichol et al., 2018); Meta-SGD also learns update directions and learning rates (Li et al., 2017). Learned optimizers parameterize update behavior (Andrychowicz et al., 2016), and self-training learned optimizers can train one another within a population (Metz et al., 2021). AutoML-Zero searches over learning algorithms assembled from elementary operations (Real et al., 2020). These are substantive antecedents to improving learning machinery. They also show why an editable or learned update policy should not automatically be described as unrestricted recursive self-improvement.

Gödel machines formalize self-modification under proof-based improvement conditions (Schmidhuber, 2007). STOP evaluates an improver that recursively operates on its own code through downstream utility (Zelikman et al., 2024). The Darwin Gödel Machine evolves a population of coding agents (Zhang et al., 2025), and Hyperagents explicitly makes task and meta-agent implementations editable (Zhang et al., 2026a). AEvo edits mechanisms governing later evolutionary search (Zhang et al., 2026b); HSI separates a task harness, an evolver, and a meta-evolver under a frozen outer anchor (Zhou, 2026). These papers constrain any claim based only on a hierarchy of mutable components.

Table 30: Evaluation controls documented in nearby work and their relation to the present repairer study.

Work	Relevant control or mechanism	Boundary of comparison
Hyperagents	Frozen-meta improvement evaluation; transfer	Transfer includes task and meta code
STOP	Downstream utility; held-out tasks; transfer	Not a dynamics-repair benchmark
CHILL-Harness	Paired branches from restored checkpoints	Shared continuation and accounting rules; branch-specific costs
SkillMentor	Frozen executor; matched skill infrastructure	Learned Mentor and skills, not the present repair state
Lean agent	Immutable verifier; held-out proof evaluation	Proof correctness differs from dynamics identification
AutoMem	Search/validation/test separation	Architecture search and diagnostic feedback

Recent methods also vary the inherited evidence and selection objective. MetaSkill-Evolve distinguishes task and slower meta-skills (Wang et al., 2026). MENDEL uses comparative evidence across tasks and lineages (Liu et al., 2026); the Huxley–Gödel Machine considers descendant productivity (Wang et al., 2025). A self-modifying Lean agent evolves proof and repair workflows under trusted verification (Li et al., 2026), and TTHE changes executable harnesses using test-time traces while holding model weights fixed (Nie et al., 2026). By comparison, Self-Refine revises outputs through feedback (Madaan et al., 2023), and Reflexion retains verbal feedback for future attempts (Shinn et al., 2023). These distinctions motivate specifying the inherited state and showing which changes persist after a reset.

Diagnosis is already a learnable object. Model-based diagnosis of planning failure uses expectations and a planner model to identify faulty assumptions (Birnbaum et al., 1990). Introspective multistrategy learning constructs learning strategies through reasoning about failures (Cox & Ram, 1999); reflective introspective learning is a further antecedent (Fox, 1998). Recent neuro-symbolic systems integrate logical failure detection with recovery planning (Cornelio & Diab, 2024), and abductive reflection detects and rectifies inconsistency in neural outputs (Hu et al., 2025). ANNEAL makes symbolic patches explicit and governs their acceptance (Hakim et al., 2026). These approaches motivate a distinction between diagnosing a task failure, repairing a model, and changing future diagnosis.

SkillMentor directly trains diagnosis and skill curation while freezing the executor, and compares learned and untrained mentors with matched skill infrastructure (Bao et al., 2026). Therefore even component isolation of learned diagnosis has prior art. AutoMem localizes memory failures to modules and validates whether diagnostic directions improve architecture search (Du et al., 2026). CausalFlow intervenes on execution steps and reruns dependent continuations to assess repairs (Bonagiri et al., 2026). Deep Adaptive Design learns a policy that chooses experiments from prior observations (Foster et al., 2021). In all these comparisons, history-dependent behavior can change while a controller remains fixed; evidence of changed behavior must be separated from evidence of changed procedural state.

Attribution and evaluation limits. CHILL-Harness already uses counterfactual harness learning (Fu et al., 2026). Hyperagents evaluates frozen meta-agents and transfers implementations; STOP assesses held-out downstream improvement. Consequently, our common-start experiment is a particular application and combination of controls, not evidence that counterfactual procedural evaluation was previously absent. Dream-RSI changes exploration-policy code through discovery-tree replay (Zheng et al., 2026); replayed support differs from an executable physical transition model. HELIX makes harness interventions and provenance auditable, but its inspected abstract reports one evolution round (Fan & Huang, 2026). SGM uses statistical tests and error spending to gate successive edits (Wu et al., 2026). Neither an immutable evaluator nor repeated acceptance testing is novel by itself.

Selection and repeated evaluation introduce familiar statistical risks. Optimizing a noisy selection criterion can yield optimistic estimates (Cawley & Talbot, 2010); nested evaluation separates tuning from assessment (Varma & Simon, 2006). The optimizer's curse explains why the selected maximum can disappoint (Smith & Winkler, 2006). Adaptive holdout methods and reusable leaderboards address information revealed through repeated evaluation (Dwork et al., 2015; Blum & Hardt, 2015). These sources motivate freezing evaluation access; they do not confer their formal guarantees on our finite experiment. Comparisons across datasets and repeated resamples require attention to dependence and the unit of replication (Demšar, 2006; Dietterich, 1998; Nadeau & Bengio, 2003).

Counterfactual risk minimization and doubly robust evaluation provide related treatment of logged feedback (Swaminathan & Joachims, 2015; Dudík et al., 2011; Jiang & Li, 2016). Their assumptions must not be silently imported: a paired replay on a fully specified simulator is not automatically an unbiased off-policy estimator. Program repair likewise supplies useful precedents without proving procedural improvement. GenProg and automatic patch generation search for changes accepted by tests (Le Goues et al., 2012; Weimer et al., 2009); SemFix and Angelix use semantic or symbolic constraints in repair (Nguyen et al., 2013; Mechtaev et al., 2016). Passing a fixed suite can validate a candidate within that suite while leaving generalization and the quality of the repair procedure unresolved.

Access and inference boundaries. The accompanying ledger resolves identities and records inspected manifestations, dates, access levels, and narrow supported claims. The core comparisons use selected method sections; several contextual references and newer leads have only primary abstract-level support. The ANNEAL comparison uses arXiv v1; the later v2 was not inspected. This is a bounded update of supplied bibliographies, not a systematic review, reproduction, or comprehensive correction/retraction audit. Inaccessible records remain unresolved leads rather than negative evidence. We report no cross-paper performance ranking. The defensible contribution is whatever the present controlled interventions establish about inherited repair state under the stated representation, task distribution, and budget; broader recursion, representation discovery, and domain transfer remain separate hypotheses.

H A PRACTICAL GUIDE TO GROUNDED DIAGNOSIS AND REUSABLE REPAIR

This appendix specifies prospective extensions, not additional experiments. Algorithms G1 and G2 were not implemented or evaluated. The tables define interfaces and evidence requirements; Appendix F supplies the complementary attribution conditions.

H.1 INTERFACES THAT MAKE NEURAL AND SYMBOLIC COMPONENTS CONSEQUENTIAL

An executable hybrid model needs explicit neural/symbolic contracts (Table 31). One prospective architecture maps observations to uncertain object states, predicts action-conditioned changes through typed operators, and models unresolved dynamics with a neural residual. Another uses learned predicates to connect observations to abstract planning operators, as explored by VisualPredicator (Liang et al., 2025). Neural probabilistic logic and learned grounded concepts supply related representation mechanisms (Manhaeve et al., 2018; Mao et al., 2019); they do not by themselves specify a self-modifying repair procedure.

Declare whether each neural output is a state estimate, predicate probability, effect distribution or ranking score. These meanings are not interchangeable: confidence need not be calibrated, and latent coordinates need not preserve object identity. A predicate such as *reachable* changes meaning if its action set, horizon or collision model changes.

Consider a transition prediction formed by a symbolic update plus a learned residual. Good combined accuracy can conceal a wrong symbolic term because the residual cancels it. Conversely, an exact symbolic constraint may conceal poor neural grounding if violations are projected back into a feasible set. Record predictions before and after correction or projection. Test interventions that change the suspected mechanism while preserving relevant nuisance factors. Without such tests, a successful repair may merely move error between components.

WorldCoder and PoE-World provide inspectable model revision and composition (Tang et al., 2024; Piriyakulkij et al., 2025). Check one-step prediction, multi-step rollout, action ranking and con-

Interface	Contract to declare	Failure and diagnostic evidence
Observation to state	Object identities, units, missingness, uncertainty, and temporal alignment.	A sensor shift appears as a dynamics error; compare independent observations and identity consistency.
State to predicate	Grounding function, threshold or probability semantics, and unknown state.	A threshold hides counterexamples; inspect continuous scores near the boundary.
Operator to neural effect	Input domain, coordinate frame, action encoding, and predicted quantity.	A residual compensates for a wrong rule; inspect uncorrected and combined predictions.
Model to planner	Horizon, admissible states, termination, rewards, and constraint semantics.	Planning exploits unsupported states; record visited simulated states and support diagnostics.
Patch to dependent modules	Changed symbols, dependencies, migration rule, and invalidated caches.	A valid local edit changes another module's meaning; execute dependency-specific checks.

Table 31: Prospective interface contracts. Passing a type check verifies only the declared software contract, not that its semantics match the environment.

straint satisfaction separately. An operator may fail when another changes its preconditions. Neural-only and symbolic-only comparisons must declare resulting changes in information, representation and expressivity.

H.2 Diagnosis as a falsifiable account of a failure

Diagnosis should produce competing explanations and predicted consequences (Table 32). Historical work on model-based planning diagnosis and introspective learning motivates this distinction (Birnbaum et al., 1990; Cox & Ram, 1999; Fox, 1998). A useful record contains the affected component, observed discrepancy, alternative hypotheses, evidence that would distinguish them, and conditions under which no distinction is possible.

A motion error, for example, could reflect incorrect force dynamics, a changed sensor scale, an omitted external disturbance, or planning outside the model's observed regime. Each suggests different evidence. Repeating the same rollout may reduce measurement noise while leaving these explanations observationally indistinguishable. Additional observations count as diagnostic progress only if they reduce a relevant ambiguity or improve decisions. This motivates reporting abstention and unsuccessful probes as outcomes rather than treating every failure as a mandatory patch opportunity.

Hypothesis	Discriminating observation or intervention	Evidence that remains insufficient
Wrong action effect	Compare predicted response to varied actions in a supported state region.	Error reduction on a single repeated action.
Misgrounded predicate	Inspect raw observations around the predicate boundary using an independent label where available.	A downstream plan succeeds after changing the threshold.
Omitted exogenous process	Repeat under controlled or recorded external conditions; assess remaining stochastic variation.	A deterministic patch fits one disturbance realization.
Invalid composition	Test each fragment and their ordering; measure intermediate states where feasible.	Both fragments pass isolated unit tests.
Planner exploitation	Evaluate chosen rollouts against the independent environment and compare their support.	High return predicted by the repaired model itself.

Table 32: Prospective diagnostic hypotheses. Independent evidence may be unavailable; in that case the explanation remains unresolved rather than becoming ground truth.

Injected defects can label localization tests, but labels must not leak through filenames, error codes or diagnostic features. Observational scientific data may admit several adequate explanations without a privileged true defect. Test their distinguishing predictions; agreement between two generated explanations is not evidence of cause.

Abstention can retain a model, restrict planning, request a measurement or refer a case for external review. Define its action and utility. A procedure that avoids every difficult case can have few incorrect repairs yet be unusable; report coverage, conditional repair error and downstream decision performance together.

Prospective algorithm G1: evidence-seeking diagnosis and local repair

1. Read a permitted failure trace and the model's interface contracts. Produce several typed explanations, including measurement error and model inadequacy where relevant.
2. For each explanation, predict a discriminating observation and the direction of change expected from a local patch. Record a plausible competing explanation.
3. Select an admissible probe or abstain when no available probe distinguishes the consequential alternatives. Acquire the corresponding external observation.
4. Revise the explanations using that observation. Preserve unresolved alternatives instead of forcing a single diagnosis.
5. Propose a scoped model patch with expected effects on prediction, planning, and dependent modules. Check software validity, interface compatibility, and declared development probes.
6. Commit, defer, or reject the patch; record observed effects and disagreement with the diagnostic prediction. Preserve this discrepancy for later procedural learning.

This specification supplies no optimal-design or correctness guarantee. ANNEAL offers a neighboring typed patch pipeline (Hakim et al., 2026); choosing discriminating explanations and detecting failure of the pipeline remain separate decisions.

H.3 Applicability and composition of retained repairs

A reusable repair needs an applicability domain and preserved properties (Table 33). DreamCoder illustrates how learned abstractions can support later program search (Ellis et al., 2020). For world-model repair, a retained transformation might add a missing action precondition, change a coordinate convention, or split a mechanism according to context. Such transformations can transfer only if their semantic prerequisites hold.

Express applicability through observable or testable properties, without requiring the true defect. Retain negative examples and uncertainty. Specify supported states, actions, horizons and measurement conditions: average improvement can conceal harm in a rare regime.

Track dependencies between abstractions and their grounding. A changed encoder or unit convention can invalidate an unchanged symbolic name. Revalidate affected applicability records and dependent modules; keeping stale fragments active can increase ambiguity and search cost.

Withhold compositions, not only new draws of familiar arrangements. Include interfering patches, test whether applicability checks detect their conflict, and assess fallback behavior. Keep a repair narrow when its evidence supports only a restricted semantic contract.

Retained object	Minimum associated information	Challenge before broad reuse
Grounded predicate	Observation mapping, unknown-value semantics, and calibration domain.	New views, sensors, and states near its decision boundary.
Dynamics fragment	State/action signature, units, support region, and dependencies.	New parameters and withheld combinations of known mechanisms.
Repair transformation	Preconditions, changed fields, predicted effects, and inverse or recovery operation.	Negative examples and cases with plausible competing diagnoses.
Composition rule	Ordering, shared variables, preserved invariants, and conflict resolution.	Order reversal, simultaneous effects, and interacting boundary cases.
Applicability test	Evidence used, uncertainty, rejection behavior, and validation history.	New regimes where the original diagnostic cue ceases to be informative.

Table 33: Prospective records for reusable abstractions and repairs. These records support inspection and targeted testing; they do not certify general validity.

Prospective algorithm G2: promotion and retirement of repair transformations

1. Collect successful and failed scoped repairs with their diagnostic predictions, observation conditions, and dependency versions.
2. Propose a reusable transformation and an applicability test from these records. Keep examples used for construction distinct from examples used to challenge reuse.
3. Check its type signature and preserved interfaces. Execute it on compatible positive cases, plausible negative cases, and compositions with existing transformations.
4. Record where predicted benefits fail, where the applicability test errs, and where recovery is needed. Restrict the transformation or retain it as provisional if evidence is insufficient.
5. Promote the validated version with explicit dependencies and expiry conditions. Invalidate dependent records when grounding, schemas, or dynamics change.
6. Retire or revise transformations whose prerequisites no longer hold. Preserve their failure records so that later proposals need not repeat the same unsupported generalization.

A fixed learner or an editable procedure could execute this library process. Its success alone establishes neither generator self-revision nor the incremental benefit of permitting generator edits.

H.4 Extending to scientific decisions with measured outcomes

Table 34 distinguishes scientific evidence settings. In a finite molecular pool, selecting a compound reveals an existing assay label. Traceable provenance can establish that the label was measured. The selection sequence remains retrospective; it creates neither a new physical experiment nor a dynamical model of chemistry.

A practical measured-data extension should preserve assay identity, units, replicate handling, censoring, missingness, and collection time where available. Missing labels remain missing; model predictions can guide selection but cannot be silently substituted as measured outcomes. Standardizing measurements from different protocols requires justification. Splits should challenge the intended generalization—for example, new chemical series or later collection periods—while preventing near-duplicate compounds or shared experimental batches from making transfer appear easier than it is.

Repair could target uncertainty, observation models, applicability or acquisition. Executable chemistry additionally requires a specified state, intervention and transition supported by observations. Property prediction, docking, assay activity and therapeutic benefit remain distinct endpoints. Prospective work needs instrument constraints and independent measurement checks.

Evidence setting	What an action can reveal	Claim requiring additional evidence
Finite measured property pool	An available label for the selected candidate under its recorded assay conditions.	Outcomes for unmeasured compounds or changed experimental conditions.
Recorded reaction experiments	A logged outcome for a documented substrate, condition, and protocol.	An unlogged intervention or a history-dependent reaction trajectory.
Validated dynamical simulator	A synthetic transition for an admissible state and action.	Physical validity outside the simulator's assessed domain.
Prospective instrument loop	A newly collected outcome for an actually executed, documented experiment.	Generalization to untested instruments, regimes, or biological endpoints.

Table 34: Prospective scientific validation settings. Each supports a different claim; synthetic and measured observations should remain visibly distinguished.

H.5 A STAGED RESEARCH PROGRAM

Begin by testing whether the hybrid representation permits meaningful, diagnosable failures and whether the planner benefits from correct repairs. Next remove privileged defect information and test competing explanations, discriminating probes, and useful abstention. Then evaluate reusable transformations across new parameterizations and withheld compositions, including deliberate applicability failures. Only after these components behave coherently should a larger editable generator be credited with constructing them autonomously.

A fixed diagnostic learner may suffice, an executed generator may add no benefit, and a repair may transfer only narrowly. Measured or prospective extensions should test these boundaries using traceable outcomes and domain-specific controls.