# WHAT CHANGES, WHEN? INTERVENTION EFFECTS ALONG CODING-AGENT TRAJECTORIES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Interactive agents act through trajectories rather than single decisions. We ask whether the effect of an intervention is stable across that trajectory, or instead depends on what is measured and when it is measured. Across 1,404 randomized coding-agent trajectories, shifting the communicated objective from speed toward reliability roughly doubles observable verification activity, showing that process behavior is highly intervention-sensitive. Controlled-error tasks then reveal that greater evaluation opportunity can be followed by markedly different repair behavior, motivating experiments that physically separate evaluation from repair.

In these stage-separated experiments, the candidate is immutable while agents receive one or three evaluation opportunities; editing then opens under a common repair interface and common downstream action allowance. Assignment to greater pre-repair evaluation opportunity shifts the first repair-stage response toward immediate correction by **53.5 percentage points**, and a separately collected cohort with a longer repair continuation shows a **62.5-point** shift. Follow-up replay reproduces much of this reallocation when substantive diagnostic content is present, narrowing the interpretation toward information content rather than quota display or redundant repetition.

In a historical paired analysis, holding the outcome definition fixed, the treatment effect on the same repair-locus correctness variable is **53.1 points** at the first repair checkpoint but only **6.25 points** at termination, a direct early-to-terminal contraction of **46.9 points**. In a separate randomized objective-framing study, the same RELIABILITY-versus-SPEED contrast yields a **27.4-point** response on an earlier event-based verification measure but only **6.1 points** on a terminal explicit-VERIFY measure.

These experiments show that intervention effects in interactive agents can be localized, transient, and response-dependent. A causal effect observed at one point in a trajectory need not describe what remains later, and a broad behavioral label need not correspond to a unique treatment response. Studying interventions on interactive agents therefore requires resolving both the response object and its location along the trajectory.

## 1 INTRODUCTION

Coding agents act along extended trajectories. They inspect programs, execute tests, form hypotheses, edit code, observe the consequences, and continue acting. An intervention can therefore affect several parts of the same interaction: how much evidence the agent gathers, when it stops checking, what it does when editing becomes possible, the state reached after an early action, and the final artifact.

Most evaluations compress this evolving process into a small number of measurements. Terminal correctness answers whether the task was ultimately solved. Process-oriented evaluations retain more of the trajectory, but they too usually summarize behavior using a particular event count, tool choice, or checkpoint. These measurements are individually meaningful, yet they need not carry the same treatment response. A large early effect can shrink as later interaction unfolds, and two plausible measurements of what appears to be the same behavior can respond differently to the same randomized intervention.



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.

We study this structure through randomized experiments on coding agents. We begin with a broad manipulation of communicated objective. Across 1,404 trajectories, the underlying technical task is held fixed while agents are instructed to prioritize speed, remain neutral, or prioritize reliability. Reliability-oriented agents perform roughly twice as much observable verification as speed-oriented agents. The aggregate direction is positive across all three served model families and all three wording variants, while task-level responses are more heterogeneous. Verification behavior is therefore not a fixed property of a model-task pair; the objective under which an agent acts can substantially reorganize its trajectory.

The consequences of this flexibility became more striking in controlled-error tasks. In one representative task, an exploratory analysis showed substantially worse downstream repair under the high-opportunity condition together with a concentration of destructive first repair actions. Some trajectories had already identified the relevant error before repair and nevertheless damaged the candidate once editing became available. This observation motivated a narrower question: *can changing what happens before repair causally change how the agent acts after repair begins?*

We therefore separate evaluation from repair experimentally. During evaluation, the candidate is immutable and agents receive different evaluation opportunities. The differential allowance then ends; the candidate becomes writable and both conditions enter the same repair interface with the same downstream action allowance. Assignment to greater pre-repair evaluation opportunity produces a large reallocation of the first repair-stage response: agents move from further checking toward immediate correction by $+53.5$ percentage points in one prospective cohort and $+62.5$ points in a separately collected cohort with a longer continuation. Follow-up replay reproduces much of this shift when substantive diagnostic content is present, while simpler quota-display and redundant-repetition manipulations do not.

This localization raises the central question of the paper: *how stable is an intervention effect as the trajectory continues?* In a historical paired analysis of a stage-separated cohort, we hold the outcome object fixed and measure the same repair-locus correctness variable at two points. The treatment effect is $+53.1$ points at the first repair checkpoint but only $+6.25$ points at termination; the directly estimated early-minus-terminal difference is $+46.9$ points. The response definition is unchanged, so the randomized treatment contrast itself contracts as the trajectory unfolds.

A separate randomized framing study reveals a complementary form of dependence. On the same eligible trajectories, the RELIABILITY-versus-SPEED intervention produces a $+27.4$-point response on an earlier event-based verification measure but only $+6.1$ points on a terminal explicit-VERIFY measure. Both describe verification-related behavior, yet they correspond to materially different treatment responses.

Together, these experiments show that intervention effects in interactive agents can be **localized**, appearing strongly at particular points in a trajectory; **transient**, shrinking as later interaction proceeds; and **response-dependent**, changing with the behavioral object used to measure them. Our contributions are empirical:

1. We establish broad randomized context sensitivity of coding-agent verification behavior across a multi-task, multi-model panel.
2. We causally localize an upstream evaluation intervention to downstream action allocation and show, on the same correctness object, a large early treatment effect that contracts sharply by termination.
3. We show that the same randomized objective intervention yields substantially different responses under two reasonable verification-related operationalizations.

These findings connect intervention claims to both *what response is measured* and *where along the trajectory it is measured*.

## 2 Experimental Setting and Study Design

Our experiments use coding agents interacting with executable environments while repairing deliberately faulty programs. Each trajectory records instructions, tool use, execution results, candidate modifications, and final artifacts. The studies manipulate either the objective under which the agent

Table 1: Exploratory adverse repair anatomy in the controlled-error task that motivated stage separation. Percentages are within condition.

| Response | Low opportunity | High opportunity |
|---|---|---|
| Final correct | 58/64 (90.6%) | 36/64 (56.3%) |
| Candidate modified | 63/64 (98.4%) | 56/64 (87.5%) |
| Correct among modified | 58/63 (92.1%) | 36/56 (64.3%) |
| Destructive first repair action | 1/64 (1.6%) | 12/64 (18.8%) |

operates or the opportunity it has to evaluate the candidate before and during repair. They expose different parts of the same interaction process and are analyzed as distinct randomized contrasts rather than pooled into one treatment effect. Numerical magnitudes are interpreted within study: we do not treat effect sizes from different cohorts or intervention families as estimates of a common effect unless a direct paired contrast is defined.

### 2.1 Objective framing and verification behavior

The broad framing experiment spans 13 coding tasks, three independently served model families, three wording variants, three communicated objectives (SPEED, NEUTRAL, RELIABILITY), and four repetitions per cell, for $N = 1{,}404$ trajectories. Its primary behavioral measure, M1, is deliberately operational: a non-malformed `execute_code` action contributes one event when its code matches at least one of 15 frozen regular-expression rules. An action matching several rules still contributes only one M1 event. The rules cover explicit assertions and equality checks, test/check definitions and named test frameworks, generic validate/verify calls, and a small number of task-relevant property checks; the complete frozen list is reproduced in Appendix E.1.1. This definition measures observable verification-like execution rather than latent intent. Removing each frozen rule in turn leaves the RELIABILITY–SPEED effect positive in all 15 variants.

A separate objective-framing study crosses SPEED versus RELIABILITY with verification cost across six tasks and two served configurations. Of 240 assigned trajectories, 238 contain a valid terminal decision. For these trajectories we retain two verification-related responses: whether an observable verification event occurred during an earlier window, and whether the agent explicitly selected `VERIFY` at the terminal decision. Because both responses are available on the same eligible trajectories, their treatment responses can be compared directly.

### 2.2 Controlled-error discovery and the transition to repair

A controlled-error factorial study contains six tasks, two model families, and 768 valid trajectories, varying evaluation opportunity together with a generic versus diagnostic checking orientation. Its aggregate outcome branches are not used as headline evidence here. The study is important because its trajectory anatomy exposed a sharper phenomenon: in one task, greater evaluation opportunity coincided with markedly worse repair and a concentration of destructive first repair actions. All 12 destructive first actions observed in the high-opportunity arm ended incorrect, and some trajectories had already localized the relevant error upstream before entering repair. This exploratory anomaly motivated a design in which evaluation and repair are separated explicitly.

### 2.3 Stage-separated evaluation and repair

The stage-separated design contains two phases. During **evaluation**, the candidate is immutable; agents may inspect the task and execute code but cannot modify the candidate. LOW receives one accepted execution opportunity and HIGH receives three. The complete interaction history is retained. During **repair**, the candidate becomes writable and both arms receive the same repair instruction, tools, and downstream action allowance within a cohort. The repair prompt does not restate the treatment condition. The treatment therefore identifies a contrast between two assigned evaluation-opportunity regimes; it does not estimate a per-execution dose-response function.

We collect two prospective cohorts on the same eight-task, three-configuration panel. The shorter continuation contains 288 assignments and provides two downstream action credits. The longer continuation contains 192 assignments and provides six. The main behavioral response is the first

Table 2: Experimental program and evidential roles. Internal project labels are omitted; each row corresponds to a distinct analysis.

| Study | Randomized contrast | Primary response | Role |
|---|---|---|---|
| Objective framing | SPEED / NEUTRAL / RELIABILITY | Verification-event count | Establish behavioral context sensitivity |
| Controlled-error discovery | Evaluation opportunity $\times$ checking orientation | Repair trajectory anatomy | Motivate cross-stage question |
| Stage-separated opportunity | 1 vs. 3 pre-repair evaluations | First writable action | Causally localize downstream action reallocation |
| Same-object temporal analysis | Same opportunity contrast | Repair-locus correctness | Compare early and terminal effects on one outcome |
| Matched behavioral definitions | RELIABILITY vs. SPEED | Earlier event vs. terminal `VERIFY` | Compare operationalized treatment responses |

repair-stage response, coded by the semantic orientation of the first emitted action attempt when one exists, with no-action retained as a separate category. Two independent blinded human coders classify responses as correction-oriented, further evaluation, another valid action, no action, or unclassifiable. Binary correction-oriented coding agrees on all 288 records in the two-credit cohort and all 192 records in the six-credit cohort. Agreement on the finer five-category labels is 269/288 (93.4%, $\kappa = .883$) and 179/192 (93.2%, $\kappa = .878$), respectively; disagreements are confined to evaluation-versus-other distinctions and do not change the primary binary endpoint.

We also score candidate correctness after the first accepted repair action and again at termination. These state-based outcomes are distinct from the semantic first-action response: an agent may attempt a correction without immediately producing a correct candidate, or may continue evaluating before later reaching a correct state.

### 2.4 Same-object temporal comparison

To examine how a treatment effect changes as the trajectory unfolds, we rescore a historical stage-separated cohort on two intact controlled-error tasks. The underlying intervention again varies pre-repair evaluation opportunity and provides a common two-action repair continuation. The response object is held fixed: the same repair locus is scored immediately after the first repair credit and again at termination using the same correctness definition. The analysis contains 128 trajectories, equally divided between treatment conditions.

### 2.5 Estimation

Within each study we preserve its randomized contrast. For fixed-panel stage-separated experiments, binary treatment effects are percentage-point differences with equal weighting across task-by-configuration strata. The same-object analysis directly estimates the difference between early and terminal treatment effects on the same trajectories. The matched behavioral comparison uses a stratified matched-block bootstrap that retains dependence between the two endpoints. Verification-count effects in the broad framing experiment are summarized with incidence-rate ratios. The two headline comparisons are direct difference estimands; they are not inferred from one endpoint being statistically significant while another is not. Where secondary correctness endpoints are treated as a testing family, Appendix C reports the corresponding multiplicity-adjusted results. Effect magnitudes are not numerically compared across unrelated cohorts as if they shared one estimand. Appendix C gives the corresponding estimators and intervals.

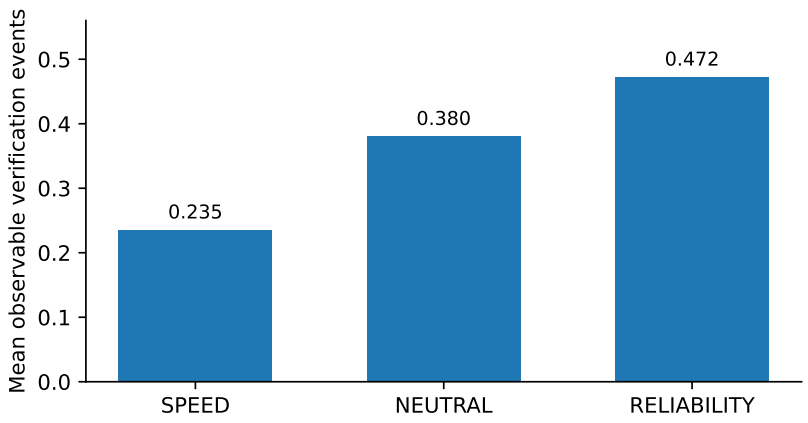



Figure 1: Communicated objective reorganizes observable verification behavior. Bars show mean M1 event counts; the randomized RELIABILITY-versus-SPEED incidence-rate ratio is approximately 2.01 $[1.60, 2.53]$.

# 3 RESULTS

## 3.1 COMMUNICATED OBJECTIVES REORGANIZE VERIFICATION BEHAVIOR

Holding the technical task fixed, communicated objective produces a large behavioral response. Mean observable verification activity rises from approximately $0.235$ events per trajectory under SPEED to $0.380$ under NEUTRAL and $0.472$ under RELIABILITY. The RELIABILITY-to-SPEED incidence-rate ratio is approximately $2.01$ with 95% CI $[1.60, 2.53]$. The three served model-family contrasts and all three wording-specific contrasts are positive; task-level effects are more heterogeneous, with 12 of 13 descriptive task contrasts positive. A frozen-rule sensitivity removes each of the 15 M1 regexes one at a time: all 15 derived endpoints remain positive, and the most attenuated retains 80.18% of the original RELIABILITY–SPEED mean difference. A pre-specified strict subset containing only explicit assertions/equalities, test/check definitions, and `unittest`/`pytest` retains 94.60%. Magnitudes nevertheless vary substantially across served model families (IRRs approximately 1.49–3.24; reported $I^2 \approx 85\%$). A separate prospective replication on 12 BigCodeBench tasks (144 planned trajectories, 142 valid) yields a task-equal RELIABILITY-minus-SPEED M1 contrast of $+0.408$ with 95% CI $[0.256, 0.561]$ ($p = 1.03 \times 10^{-4}$, task as the inferential unit). The pooled R3 IRR is therefore a strong within-study summary, not a common effect shared by every task or model configuration.

This result establishes the starting point for the rest of the paper: an agent's process behavior can move substantially while the underlying technical task remains fixed. The controlled-error anomaly in Table 1 then suggests that such upstream differences can be followed by qualitatively different repair trajectories.

## 3.2 UPSTREAM EVALUATION REORGANIZES THE FIRST REPAIR ACTION

Stage separation turns that suggestion into a direct randomized test. In the two-credit continuation, correction-oriented first attempts occur in 30/144 LOW trajectories (20.8%) and 107/144 HIGH trajectories (74.3%), an adjusted difference of $+53.47$ percentage points with 95% CI $[44.89, 62.06]$. In the independently collected six-credit continuation, the corresponding rates are 8/96 (8.3%) and 68/96 (70.8%), a difference of $+62.50$ points $[53.58, 71.42]$. The longer cohort serves as a horizon-robustness replication of the reallocation: the two cohorts are interpreted separately rather than as a randomized two-versus-six-credit comparison.

The longer continuation shows that the first-action pattern is not a consequence of giving the agent only two downstream actions. The result instead localizes a large treatment response at the transition where editing first becomes possible. Recorded correctness differences in these prospective cohorts are much smaller, but those contrasts involve a different response object and, in the two-credit cohort,

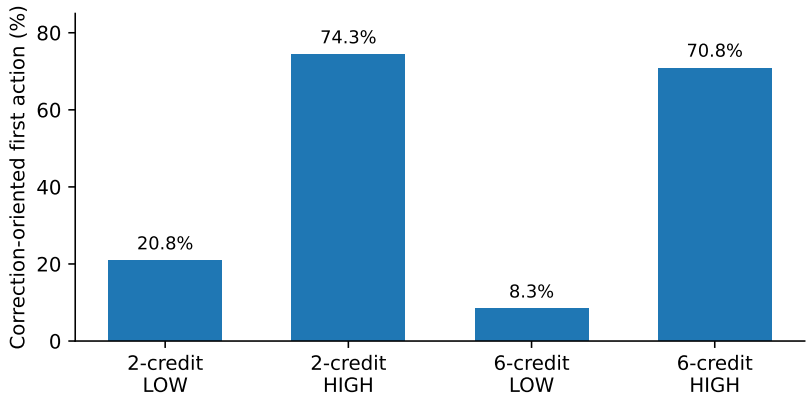



Figure 2: First repair-stage response after the common writable phase opens. Assignment to greater pre-repair evaluation opportunity strongly shifts action allocation toward immediate correction in both prospective cohorts.

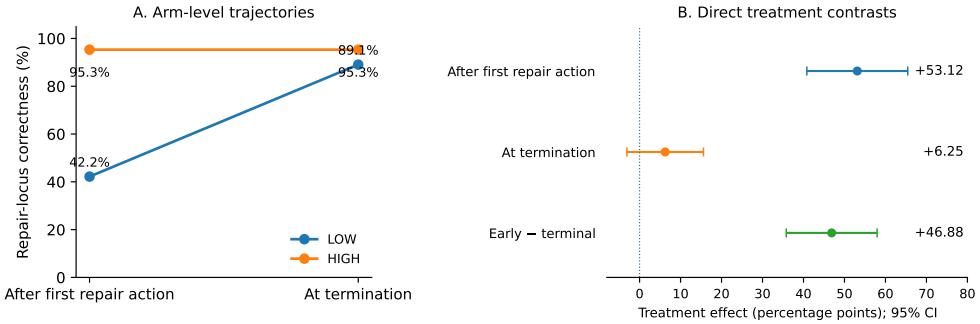



Figure 3: Same-object temporal contraction. **Left:** arm-level correctness on the same 128 trajectories: LOW rises from 42.2% to 89.1%, while HIGH remains at 95.3% at both checkpoints. **Right:** treatment effects on the identical repair-locus correctness object and their directly estimated early-minus-terminal difference. The contraction is therefore descriptively a LOW catch-up pattern at the aggregate level.

sparse successful repairs. The cleaner temporal test therefore comes from holding correctness itself fixed.

### 3.3 The same correctness effect contracts along the trajectory

On the historical intact cohort, repair-locus correctness at the first repair checkpoint is 27/64 in LOW and 61/64 in HIGH, a treatment effect of $+53.125$ points with 95% CI $[40.81, 65.44]$. At termination, the same repair-locus correctness variable is 57/64 versus 61/64, a point estimate of $+6.25$ points with 95% CI $[-3.10, 15.60]$. The directly estimated early-minus-terminal treatment-response difference is $+46.875$ points $[35.79, 57.96]$. At the arm level, the contraction has a particularly simple descriptive form: LOW rises from 42.2% correct at the first repair checkpoint to 89.1% at termination, while HIGH is 95.3% at both checkpoints. Thus the aggregate contraction is expressed as LOW catching up over the remaining interaction rather than as a decline in the HIGH arm.

The contrast therefore changes sharply even when the outcome definition does not. The large early difference mostly disappears by the end of the allowed interaction. This is the most direct evidence in the paper that an intervention effect observed at one stage need not describe what remains later.

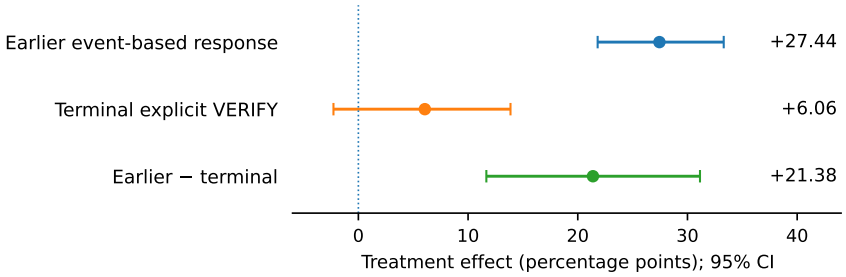



Figure 4: The same randomized objective contrast under two verification-related operationalizations. Earlier event-based behavior and terminal explicit `VERIFY` carry materially different treatment responses.

### 3.4 A BEHAVIORAL LABEL DOES NOT DETERMINE A UNIQUE TREATMENT RESPONSE

The objective-framing studies provide a complementary comparison. On 238 trajectories with valid terminal decisions, the RELIABILITY-minus-SPEED effect is $+27.44$ percentage points for the earlier event-based verification response and $+6.06$ points for terminal explicit `VERIFY`. A matched block bootstrap estimates their treatment-response difference as $+21.38$ points $[11.67, 31.14]$.

The two measurements differ in temporal coverage and behavioral form, so this experiment does not reduce the difference to time alone. Its contribution is complementary to Figure 3: the same-object analysis varies observation stage while holding the outcome definition fixed; the matched behavioral analysis shows that changing how a behavior is operationalized can also change the estimated response to the same randomized treatment.

## 4 REPLAY NARROWS THE INTERPRETATION OF THE CROSS-STAGE SHIFT

The stage-separated intervention changes the opportunity to gather diagnostic evidence before repair. Follow-up replay studies ask which visible features of that history can reproduce the large first-action shift. Displaying quota differences or adding redundant repetition does not reproduce the original positive shift. In contrast, replaying an authentic second diagnostic interaction produces a large change in later action allocation, and much of that shift remains when comparable substantive content is supplied passively.

These results narrow the interpretation toward the substantive diagnostic content available at repair entry rather than quota display or repetition alone. They do not identify a unique mediator or internal mechanism, and the information-matched comparison leaves uncertainty about any residual effect of acquisition form. A post-hoc semantic sensitivity further reduces that residual at the point-estimate level and is reported only as supporting analysis in Appendix F.

The replay evidence is therefore interpretive support for the stage-resolved result, not a separate mechanism claim. The temporal analysis asks a different question: once an intervention has created a treatment difference at the writable transition, how much of that difference remains as interaction continues?

## 5 RELATED WORK

**Verification and repair.** Self-debugging (Chen et al., 2024), learned execution critics (Tao et al., 2026), repository knowledge acquisition (Lin et al., 2026a), and execution-cost studies (Lin et al., 2026b) show that testing and feedback can change coding-agent performance and behavior. Haeri & Ghelichi (2026) distinguish specification-grounded test effectiveness from test quantity. Our experiments focus on the causal location of these responses: we randomize objective or pre-repair opportunity, observe what changes at a later writable transition, and compare those effects with later states.

**Process evaluation.** Trajectory review (Podivilov et al., 2026), failure-process analysis (Zhao et al., 2026), action/task/step process evaluation (He et al., 2026), and protocol cards (Zhang & Lee, 2026) motivate looking beyond a single end score. We build on that perspective by estimating randomized treatment contrasts at different points in a trajectory, including a direct same-outcome early-versus-terminal comparison and a matched comparison between two behavioral operationalizations. Our contribution is not the generic observation that intermediate and terminal metrics can differ, but the empirical size and structure of randomized treatment-response differences, including a 46.9-point contraction on the same correctness object.

**History, replay, and evolving state.** SpecPath studies contract-equivalent histories (Wu et al., 2026); causal replay methods intervene on agent steps (Shah, 2026; He et al., 2026); state-bound evidence work distinguishes transient from current correctness (Gao et al., 2026). Our replay experiments narrow the interpretation of a specific cross-stage action shift. The main contribution lies in the randomized stage-resolved treatment responses and their direct contrasts, rather than in replay itself.

## 6 Discussion

**Intervention effects have trajectory structure.** The experiments reveal a recurring pattern. Communicated objectives reorganize verification behavior; evaluation opportunity reorganizes the first repair-stage response; a large correctness contrast can appear at the first repair checkpoint and mostly disappear by termination; and two verification-related measurements can carry different responses to the same randomized framing. These are different manifestations of the same empirical point: in an interactive system, an intervention effect can be concentrated at a particular stage rather than uniformly expressed throughout the trajectory.

**Terminal and stage-resolved responses answer different questions.** Terminal correctness remains the appropriate response when the scientific question is whether the task is ultimately solved. Stage-resolved responses answer a different question: where an intervention changes the interaction and whether that difference persists. A large early effect that later contracts should therefore not be read as a large terminal performance effect, but neither should it be discarded as irrelevant process noise. The two measurements correspond to different estimands.

**Response definitions shape the empirical question.** The matched framing study shows a second source of variation. "Verification behavior" can refer to an event occurring during an earlier window or an explicit terminal decision, and the same randomized treatment need not affect these responses equally. A treatment-effect statement therefore becomes more precise when it names the response object and its location in the interaction. This is especially important for agent evaluations that compare interventions using a mix of terminal success, tool counts, self-reports, and intermediate decisions.

**From broad behavior to localized causal response.** The experimental sequence also illustrates how a broad behavioral observation can be refined. Objective framing first establishes that verification is highly context-sensitive. Controlled-error trajectories then reveal unusual downstream repair patterns. Stage separation isolates a causal effect on action allocation, replay narrows the interpretation toward substantive diagnostic information, and the same-object analysis reveals that the resulting treatment contrast can change dramatically as the trajectory continues. The scientific object thus narrows from "how much does the agent verify?" to "where does a randomized intervention appear, and how does that response evolve?"

**Heterogeneity and scope.** Effect magnitude is not constant across tasks or served configurations. The broad framing experiment shows substantial family-level heterogeneity; the direction is stable across the three served families and wording variants, but not across every task-level contrast. The stage-separated results summarize a fixed panel rather than a population of arbitrary coding tasks. We therefore treat pooled estimates as within-study summaries, not universal constants. Explaining which task properties predict larger or smaller responses is a separate question from establishing that the response can vary by stage and operationalization.

**Ecological validity of stage separation.** The stage-separated design deliberately prevents the natural interleaving of testing and editing during the upstream phase. This control is what makes the downstream transition identifiable: the candidate is held fixed until the randomized opportunity difference ends. The trade-off is external validity. In free-form repository work, evaluation and modification can alternate continuously, so the magnitude of the observed transition effect need not transfer unchanged. The experiment identifies a controlled cross-stage effect; reproducing its magnitude under unconstrained interleaving remains an open empirical question.

**Engineering implications.** The results suggest that agent-system evaluations should log more than final pass/fail when decisions depend on latency, verification workload, stopping rules, or when to intervene. Objective framing alone can materially change the amount of verification work performed, and stage-resolved measurements can reveal large transient differences that terminal success no longer shows. A practical evaluation can therefore pair terminal correctness with a small number of pre-specified transition measurements rather than attempting to score every intermediate state.

**Limitations.** The prospective stage-separated panel consists of controlled coding tasks and served model configurations rather than a representative repository benchmark. Successful repair is sparse in the prospective cohorts, which limits what their correctness contrasts can establish. The same-object temporal comparison is a post-hoc analysis of a historical recovered cohort, although its early and terminal labels are reconstructed consistently on the same 128 trajectories. The matched behavioral comparison conditions on terminal-decision availability and changes temporal coverage and behavioral form together. The replay studies narrow explanations of the first-action shift but do not establish a unique internal mechanism. These constraints limit generalization of magnitudes while leaving the direct within-study treatment-response contrasts intact.

## 7 Conclusion

Interactive-agent interventions can have large effects that are concentrated at particular stages of a trajectory and much smaller effects later. In coding agents, communicated objective strongly changes verification behavior; pre-repair evaluation opportunity strongly reorganizes the first repair-stage response; the effect on the same correctness object contracts from $+53.1$ points at the first repair checkpoint to $+6.25$ points at termination; and the same objective intervention yields different responses under two verification-related measurements. Together, these results show that intervention effects in interactive agents are best understood as stage- and response-resolved quantities rather than single trajectory-invariant summaries.

## AI USE STATEMENT

Generative AI tools were used for literature retrieval and synthesis, hypothesis refinement, experimental-design and methodology feedback, implementation and analysis assistance, result interpretation, figure/code assistance, and manuscript drafting and editing. The author independently reviewed the scientific claims, analyses, code, citations, and final manuscript and takes full responsibility for the work. The primary semantic labels in the prospective V2.3 and RH6 cohorts come from independent treatment-blind human coding; AI-assisted analyses are not presented as independent human validation.

## ETHICS STATEMENT

The experiments evaluate coding agents on controlled software artifacts and do not involve human subjects or personal data. The reported claims concern observable agent behavior and scored artifacts; we do not infer human-like psychological states from tool use.

## REPRODUCIBILITY STATEMENT

Appendices A–H describe the experimental contracts, response rules, estimators, and sensitivity analyses. The supporting-materials release assembled with this manuscript includes frozen assignment-level outputs and analysis code for the prospective V2.3 and RH6 cohorts, the R3 objective-framing experiment, the C2 matched operationalization analysis, the M1 rule-sensitivity analysis, and the external BigCodeBench replication; it also includes the prospective human-coding labels and adjudication records and the recovered files used for the historical same-object analysis. These materials support regeneration of the reported numerical summaries from frozen data. Full end-to-end replay of every historical provider interaction is not claimed, and the later replay series remains supporting interpretation rather than a fully human-validated mechanism analysis.

## REFERENCES

Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. Teaching large language models to self-debug. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2304.05128.

Xueping Gao, Jianwei Yang, and Qiang Yang. Looping is not reliability: State-bound evidence and typed revision contracts for agentic code repair, 2026. URL https://arxiv.org/abs/2607.24604.

Amin Haeri and Mahdi Ghelichi. Specification grounding drives test effectiveness for LLM code, 2026. URL https://arxiv.org/abs/2607.06636.

Jiawei He, Mengyu Shi, Jie Jia, Xikai Yang, and Dong Sun. What process evaluation of coding agents actually measures: Action, task, and step are three different levels, 2026. URL https://arxiv.org/abs/2608.22960.

Haotian Lin, Silin Chen, Xiaodong Gu, Yuling Shi, Chengxi Pan, Jiaqi Ge, Mengfan Li, Jianghong Huang, Mengchieh Chuang, Beijun Shen, and Haibing Guan. Know before fix: QA-driven repository knowledge acquisition for software issue resolution, 2026a. URL https://arxiv.org/abs/2607.11111.

Zhihao Lin, Junhua Zhu, Mingyi Zhou, Xin Wang, Zhensu Sun, Renyu Yang, David Lo, and Li Li. To run or not to run: Analyzing the cost-effectiveness of code execution in LLM-based program repair, 2026b. URL https://arxiv.org/abs/2606.26978.

Andrey Podivilov, Vadim Lomshakov, Sergey Savin, Matvei Startsev, Roman Pozharskiy, Maksim Parshin, and Sergey Nikolenko. AgentLens: Production-assessed trajectory reviews for coding agent evaluation, 2026. URL https://arxiv.org/abs/2607.06624.

Jaineet Shah. Causal agent replay: Counterfactual attribution for LLM-agent failures, 2026. URL https://arxiv.org/abs/2606.08275.

Leitian Tao, Baolin Peng, Haorui Wang, Hang Wang, Hao Cheng, Wenlin Yao, Qianhui Wu, Tao Ge, Sharon Li, and Jianfeng Gao. ExecCritic: Learn to test, test to improve for coding agents, 2026. URL https://arxiv.org/abs/2609.09133.

Yangfan Wu, Haozhe Wang, Huanyu Yang, Jianmin Ji, and Fangzhen Lin. SpecPath: Testing coding agents across contract-equivalent specification histories, 2026. URL https://arxiv.org/abs/2608.09799.

Keqin Zhang and Sharon Lee. Beyond leaderboards: Protocol cards for trustworthy coding-agent evaluation. In *KDD 2026 Workshop*, 2026. URL https://kdd-eval-workshop.github.io/agenticai-evaluation-kdd2026/assets/papers/26_Beyond_Leaderboards_Protoco.pdf.

Xiangxin Zhao, Han Li, Shuaiting Li, Tianyi Zhao, Earl T. Barr, Federica Sarro, and He Ye. Failure as a process: An anatomy of CLI coding agent trajectories, 2026. URL https://arxiv.org/abs/2607.09510.

Terry Yue Zhuo, Minh Chien Vu, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, et al. BigCodeBench: Benchmarking code generation with diverse function calls and complex instructions. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2406.15877.

# A PROTOCOL, TASKS, AND SERVING RECORD

## A.1 PROSPECTIVE TASK PANEL

The prospective cohorts use the same eight-task, three-configuration panel. The task admission rule requires an erroneous supplied candidate, an available evaluation operation, and a plausible direct correction. Admission was not conditioned on observed treatment-effect success. This helps isolate a meaningful choice between evaluation and modification, but is not a difficulty calibration or evidence of repository-level competence.

Table 3: Prospective task roster and deterministic repair checks.

| Identifier | Repair target | Reference check |
|---|---|---|
| CCF-DATA-01 | Grouped revenue: aggregate before rounding | Canonical CSV output |
| CCF-DATA-02 | Unmatched orders/shipment identifiers | Canonical unmatched-ID set and order |
| CCF-EXEC-01 | Numeric-column classification in CSV summary | Hidden-fixture stdout assertions |
| CCF-EXEC-02 | ISO-week boundary handling | Standard-library reference battery |
| CCF-CALC-01 | Amortization interest/principal update | Exact-decimal reference |
| CCF-CALC-02 | Allocation residual reconciliation | Exact-rational property |
| CCF-CONS-01 | Staff-assignment capacity | Feasibility and objective check |
| CCF-STRUCT-01 | Dependency-manifest edge | Schema and declared-edge check |

The grouped-revenue example in Section 2 is grounded in the supplied task contract and candidate. The candidate adds `Decimal(amount).quantize(Decimal('.01'))` to each group and rounds the group again on return; its half-up rounding therefore occurs prematurely. The supplied public fixture has west amounts 1.20 and 2.30 and east amount 4.00. Both the erroneous and corrected procedures agree on that fixture. The 0.005 example is a constructed illustration of the defect, not an observation of which diagnostic any particular subject selected. No hidden answer is supplied by the illustrative example during an experiment.

## A.2 STAGE INSTRUCTIONS AND ACCOUNTING

The V2.3 evaluation template is:

```
An existing candidate artifact is present at
{candidate_path}.
This is the evaluation phase.  You may inspect and evaluate
the candidate using the available execution tool, but the
candidate artifact cannot be modified.
You have exactly {K_eval} environment-action credits in this
phase.  Each accepted execution-tool call consumes one
credit.  When finished, return DONE.
```

LOW substitutes one and HIGH three for `K_eval`. The transition is:

```
Correction phase:
You now have exactly 2 environment-action credits to
finalize or modify the candidate artifact at
{candidate_path} if necessary.  You may use the available
file and execution tools.  Each accepted tool call consumes
one credit.  When finished, return DONE.
```

Stage 1 writes are guarded rather than prohibited only in prose. Full history is retained and unused credits cannot be transferred across stages. The transition does not repeat an arm label or force an edit. A V2.3 stage ends on `DONE`, accepted-credit exhaustion, or its four-assistant-turn ceiling. An accepted dispatched execution consumes a credit even if its runtime fails; an unaccepted request does not automatically do so. Provider-array order determines the first attempt if multiple actions are

emitted together. Reasoning text alone is not a tool action. Thus accepted capacity, attempted action, runtime success, and terminal response are different variables.

V2.3 infrastructure settings include a 90-second per-call timeout, 15-second sandbox command timeout, and 1,800-second run watchdog. These are not one shared scientific compute budget. RH6 preserves the one/three evaluation assignment, immutable candidate, retained history, task checks, and tool interface while setting a common six-credit repair allowance. Its exact assistant-turn ceiling is not recovered; the V2.3 value is not silently reused.

### A.3 Serving configurations

The V2.3 serving record identifies three historical routes: M1 requests `qwen3.7-max-2026-06-08`; M2 requests `glm-5.3-flash`; M3 requests `deepseek-v4-flash` and records `deepseek-flash` in the returned provider metadata. Their recorded configuration hash prefixes are `7d65ddb4f6a8`, `86447867b00e`, and `2057ca9b4a77`. A later non-subject protocol probe confirmed route reachability and the same request/return naming pattern; it does not verify model-weight identity.

Configuration settings are fixed within randomized strata according to the experimental record. Exact provider versions, complete decoding parameters, and weight identities remain unavailable. The route labels are not newly verified model-family identities and are not assigned to the older R3 programs by index. Generality is therefore stated over the tested configurations, not over all deployments bearing similar marketing names.

## B Response Coding and Correctness States

### B.1 First emitted attempt

Correction orientation concerns the attempted operation on the candidate, not the tool name alone. An execution request that attempts to overwrite the candidate can be correction-oriented; an execution that probes candidate behavior is evaluation. A malformed candidate-writing attempt can still be semantically classified. An unclassifiable observed attempt is different from a completed no-action response, and both are different from infrastructure nonobservation. The prospective cohorts have no unclassifiable primary records.

The V2.3 record reports two independent blinded human coders with 288/288 binary agreement. Five-category agreement is 269/288 (93.40%; $\kappa = .883259$); the 19 disagreements concern evaluation versus other action, not the primary correction indicator. RH6 reports 192/192 binary agreement and 179/192 five-category agreement (93.23%; $\kappa = .877539$); its 13 evaluation/other disagreements are adjudicated as evaluation. Labels are locked before joining treatment keys. The supporting release contains the coder outputs, adjudication records, final labels, and coding locks for both cohorts. Blinding removes direct assignment information; it does not imply that treatment is impossible to infer from response content.

Table 4: Full primary response composition; all assigned no-action records are retained.

| | V2.3 (two credits) | | RH6 (six credits) | |
|---|---|---|---|---|
| Category | LOW | HIGH | LOW | HIGH |
| Correction-oriented | 30 | 107 | 8 | 68 |
| Further evaluation | 106 | 25 | 88 | 22 |
| Other valid action | 8 | 2 | 0 | 0 |
| No action | 0 | 10 | 0 | 6 |
| Unclassifiable | 0 | 0 | 0 | 0 |
| Total | 144 | 144 | 96 | 96 |

A completed no-action outcome maps to $C = 0$ but is not assumed to be a refusal, voluntary completion, or timeout. The available endpoint summaries do not separate these causes at case

level. The zero-versus-ten and zero-versus-six no-action asymmetries are therefore retained without a psychological explanation.

### B.2 Correctness is a separate response

Table 5 preserves all correctness states. Operational composites set not-reached or unscorable values to zero for assigned-denominator analysis. They are not statements that each such candidate was observed and incorrect.

Table 5: Correctness state accounting. Unlisted state cells are zero.

| | | V2.3 | | RH6 | |
|---|---|---|---|---|---|
| Checkpoint | State | LOW | HIGH | LOW | HIGH |
| First accepted credit | Correct | 0 | 6 | 0 | 3 |
| | Incorrect | 144 | 128 | 96 | 87 |
| | Not reached | 0 | 10 | 0 | 6 |
| Terminal | Correct | 0 | 7 | 1 | 4 |
| | Incorrect | 126 | 134 | 95 | 92 |
| | Unscorable | 18 | 3 | 0 | 0 |

In V2.3, eight distinct trajectories are correct at either checkpoint, all on grouped revenue: six at the first checkpoint and seven at termination, not identical sets. RH6 first-checkpoint successes are also on grouped revenue; the terminal record additionally includes a HIGH/M3 ISO-week success. This concentration is descriptive, not a post-hoc restriction of the randomized comparison to successful tasks.

For V2.3 terminal unknown states, the lowest arbitrary binary completion is $100(7-18)/144 = -7.638889$ points, and the highest is $100(7+3)/144 = 6.944444$. Equal arm sizes within strata make this pooled arithmetic agree with the equal-stratum estimator. These are sample identification ranges under unspecified binary completion, not sampling intervals or evidence about which unknown records were actually correct. RH6 terminal states are all scorable.

## C Inference, Fixed-Panel Breadth, and Recovery Handling

### C.1 Cohort-specific frozen inference

For response $Z$ and equal stratum weights $w_s = 1/S$, the reported fixed-panel treatment contrast is

$$\widehat{\tau} = \sum_{s=1}^{S} w_s \big(\overline{Z}_{\mathrm{H},s} - \overline{Z}_{\mathrm{L},s}\big). \tag{1}$$

Write $S^2_{as}$ for the sample variance of a binary response in arm $a$ and stratum $s$, $n_{as}$ for its size, and $a_{as} = S^2_{as}/n_{as}$. For $w_s = 1/24$, the accompanying variance estimate is

$$\widehat{V} = \sum_{s} w_s^2 (a_{\mathrm{H}s} + a_{\mathrm{L}s}). \tag{2}$$

V2.3 uses arm-level Satterthwaite degrees of freedom,

$$\nu_{\mathrm{arm}} = \frac{\widehat{V}^2}{\sum_s w_s^4 \{a_{\mathrm{H}s}^2/(n_{\mathrm{H}s}-1) + a_{\mathrm{L}s}^2/(n_{\mathrm{L}s}-1)\}}, \tag{3}$$

and $\widehat{\tau} \pm t_{\nu,.975}\sqrt{\widehat{V}}$. With six per arm per stratum, the primary standard error is .0433680417 and $\nu = 124.508841$. RH6 uses the same variance with its frozen grouped-stratum convention,

$$\nu_{\mathrm{grouped}} = \frac{\widehat{V}^2}{\sum_s w_s^4 (a_{\mathrm{H}s} + a_{\mathrm{L}s})^2/(4-1)}. \tag{4}$$

Its degrees of freedom are 41.657143 for correction orientation, 9 for first-checkpoint correctness, and 10.714286 for terminal correctness. These approximate design-based intervals are not exact randomization intervals or intervals over an exchangeable population of tasks. The cohort conventions are retained rather than changed to make intervals visually comparable.

Randomization tests preserve six-of-twelve allocation within V2.3 strata and four-of-eight within RH6 strata. Both use 100,000 draws, the two-sided $|T_b| \geq |T_{\text{obs}}|$ rule, and $p = (b + 1)/(B + 1)$. Seeds are 20260909 and 20260911. Each primary correction-first $p$ equals $1/100001$. V2.3 secondary randomization $p$ values are .020840 for first-checkpoint correctness, .009080 for terminal correctness, and 1 for their direct early-minus-terminal contrast; the three-test Holm values are .041680, .027240, and 1. Its paired correctness-effect difference is $-0.6944$ points $[-3.6145, 2.2256]$. RH6 correctness randomization values are .249318 and .359886. A significant operational composite is not a semantic-correctness guarantee for unscorable states.

Table 6: Prospective estimates and reported 95% intervals, in percentage points.

| Cohort | Response | Effect | Lower | Upper |
|---|---|---|---|---|
| V2.3 | Correction-first | 53.4722 | 44.8888 | 62.0556 |
| | First-checkpoint correct | 4.1667 | 1.0472 | 7.2861 |
| | Terminal correct | 4.8611 | 1.5504 | 8.1718 |
| RH6 | Correction-first | 62.5000 | 53.5791 | 71.4209 |
| | First-checkpoint correct | 3.1250 | −.9564 | 7.2064 |
| | Terminal correct | 3.1250 | −2.0183 | 8.2683 |

### C.2 WITHIN-PANEL SCOPE

The following summaries preserve the supplied stratum and leave-out tables. They do not supply new observations. Individual strata have only six or four records per arm, so positivity at the member-mean level should not be interpreted as precise estimation of every stratum's effect.

Table 7: All primary task–configuration contrasts (percentage points).

| | 2-credit confirmation | | | 6-credit replication | | |
|---|---|---|---|---|---|---|
| Task | M1 | M2 | M3 | M1 | M2 | M3 |
| Amortization update | 33.33 | 50.00 | 50.00 | 0.00 | 25.00 | 0.00 |
| Allocation residual | 66.67 | 100.00 | 100.00 | 75.00 | 100.00 | 75.00 |
| Staff capacity | 83.33 | 50.00 | 66.67 | 25.00 | 100.00 | 75.00 |
| Grouped revenue | 33.33 | 66.67 | 100.00 | 75.00 | 100.00 | 75.00 |
| Unmatched identifiers | 33.33 | 66.67 | 100.00 | 50.00 | 100.00 | 75.00 |
| Numeric-column classifier | 0.00 | 33.33 | 33.33 | 100.00 | 100.00 | 100.00 |
| ISO-week boundary | 0.00 | 33.33 | 83.33 | 50.00 | 50.00 | 75.00 |
| Dependency edge | 33.33 | 66.67 | 0.00 | 0.00 | 75.00 | 0.00 |

Table 8: Member means and leave-one-member-out effects (percentage points).

| | 2-credit confirmation | | 6-credit replication | |
|---|---|---|---|---|
| Member | Member mean | Leave out | Member mean | Leave out |
| Amortization update | 44.44 | 54.76 | 8.33 | 70.24 |
| Allocation residual | 88.89 | 48.41 | 83.33 | 59.52 |
| Staff capacity | 66.67 | 51.59 | 66.67 | 61.90 |
| Grouped revenue | 66.67 | 51.59 | 83.33 | 59.52 |
| Unmatched identifiers | 66.67 | 51.59 | 75.00 | 60.71 |
| Numeric-column classifier | 22.22 | 57.94 | 100.00 | 57.14 |
| ISO-week boundary | 38.89 | 55.56 | 58.33 | 63.10 |
| Dependency edge | 33.33 | 56.35 | 25.00 | 67.86 |
| Configuration 1 | 35.42 | 62.50 | 46.88 | 70.31 |
| Configuration 2 | 58.33 | 51.04 | 81.25 | 53.12 |
| Configuration 3 | 66.67 | 46.88 | 59.38 | 64.06 |

### C.3 Recovery handling

V2.3 reaches 288/288 final endpoint coverage using assignment-preserving technical recovery: task, configuration, treatment assignment, and scientific identity are unchanged. RH6 contains five transport nonobservations, each resolved by selecting the first technically complete recovery for the same assignment; no replacement assignments are introduced. The reported effects therefore pertain to these executed randomized policies rather than to a hypothetical failure-free first dispatch.

One earlier V2.3 collection summary cannot be reconciled cleanly with the final selected-source accounting, so we do not use it to estimate arm-specific first-dispatch failure rates. One retained V2.3 trajectory exceeds the two-credit repair allowance; a strict-truncation sensitivity preserves the direction of the pre-correction results. These issues affect provenance and protocol-compliance claims, not the definition of the randomized treatment contrast used in the main analysis.

## D Historical Same-Object Analysis and Corrected Replications

### D.1 Historical paired same-object analysis

A16 reuses the 128 recovered trajectories from two intact original EXEC tasks (R3C-EXEC-01 and R3C-EXEC-04). The intervention grants one versus three upstream executions, followed by two repair credits. The task restriction and matched checkpoint analysis are post-hoc. Early repair-locus correctness is 27/64 versus 61/64 ($+53.125$ points, HC2 interval $[40.81, 65.44]$); terminal correctness on the same object is 57/64 versus 61/64 ($+6.25$, $[-3.10, 15.60]$). The paired difference is $+46.875$ points $[35.79, 57.96]$. Reconstruction reproduces 128/128 first-checkpoint labels and terminal repair-locus correctness agrees with full terminal correctness on all 128.

The original first-emitted semantic composition provides an additional distinction between tool labels and attempted operation: LOW has 35 evaluation, 27 direct writing, and two no-action records; HIGH has 61 direct writing, one malformed correction-oriented execution request, and two no-action records. The latter request is not another evaluation merely because its native tool is execution. These historical semantic counts are not interchangeable with V2.3's prospective human-coded counts.

### D.2 Sensitivity to recovery handling

The historical collection includes both transport recoveries and protocol-violation recoveries. To assess sensitivity to those choices, we recompute the early treatment contrast under several deliberately adverse operational codings. These are sensitivity analyses of recovery handling, not observations of otherwise unavailable correctness states or formal lower bounds on a single-dispatch causal effect.

Reported 10,000-draw stratified randomization $p$ values are approximately .00010, .00030, and .00050 for the initial-violation, transport-only-approximation, and all-first-noncomplete composites. The early contrast remains positive across these adverse recovery codings, while the historical paired analysis remains post-hoc.

Table 9: Historical intact-EXEC early response under reported recovery sensitivities. Values are percentage points; these are not temporal-contraction replications.

| Composite policy | Effect | HC2 95% CI |
|---|---|---|
| Canonical recovered-complete | 53.125 | $[40.81, 65.44]$ |
| Initial protocol violations adverse | 40.625 | $[25.60, 55.65]$ |
| Any lineage protocol violation adverse | 39.06 | $[23.75, 54.37]$ |
| Closest transport-only recovery approximation | 32.8125 | $[16.54, 49.08]$ |
| Every first-dispatch non-completion adverse | 34.375 | $[18.02, 50.73]$ |

### D.3 Controlled-error extensions

The CONS internal extension gives a canonical early contrast of $+37.5$ points. A pure-transport-favorable hostile composite is $+25.0$ $[5.16, 44.84]$, but making every first-dispatch non-completion adverse gives $-12.5$ with a wide interval crossing zero. It is therefore supportive under the executed/reasonable transport-retry setting, not evidence of universal retry-free robustness. It also cautions against a universal claim that terminal outcome changes are always negligible.

Original DATA-10 runs omitted required `items.csv`; their original zero effect is not scientific evidence about the intended task. A separate post-discovery corrected allocation has 64 assignments, 62 valid runs, and two HIGH protocol violations without replacements. Valid early success is 26/32 LOW versus 30/30 HIGH; the adjusted contrast is $+19.18$ points $[5.41, 32.95]$. Terminal success is 30/32 versus 30/30, adjusted $+6.25$ $[-2.35, 14.85]$. Treating both undefined HIGH early responses adversely yields $+12.50$ $[-3.72, 28.72]$ over all assignments. This is a qualified corrected replication, not a clean prospective confirmation and not a reason to reinstate the invalid original zero.

## E Objective Framing and Matched Behavioral Definitions

### E.1 R3: context sensitivity of observable events

R3 crosses 13 tasks, three served families, three wording variants, three objectives, and four repeats ($13 \times 3 \times 3 \times 3 \times 4 = 1{,}404$). The objectives are SPEED, NEUTRAL, and RELIABILITY. M1 denotes the frozen observable verification-event count. The frozen implementation is `extract_m1_from_trace` in `src/p4/run_qualification.py`, introduced in the recorded implementation commit `0c2f1cbf8a7ada61f877d28b69847961cc8fadb8`. It counts a non-malformed `execute_code` action once when the submitted code matches one or more of 15 frozen regular expressions; a multi-match action still contributes exactly one M1 event. Approximate means are .235, .380, and .472 respectively. RELIABILITY/SPEED IRR is approximately 2.009 $[1.598, 2.525]$. All three served-family contrasts and all three wording-specific contrasts are positive; 12 of 13 task-level descriptive contrasts are positive. Family effects range from about 1.488 to 3.235, with substantial heterogeneity ($I^2 \approx 85\%$); the pooled ratio is not a common effect for all tasks or configurations.

#### E.1.1 Frozen M1 rule set and perturbation audit

The outcome-blind sensitivity implementation reproduces the 15 source regexes verbatim and aborts if its local list differs from the frozen `_VERIF_KEYWORDS`. It loads exactly 1,404 ledger-authoritative identities, verifies ledger/manifest identity equality and raw-trace hashes, and reproduces frozen M1 trajectory by trajectory before constructing any derived endpoint. One frozen regex is one leave-one-rule-out unit because the original implementation contains no higher-level grouping. Rule-family contribution totals are non-additive when an action matches several regexes, while M1 itself counts that action only once.

Table 10: The 15 frozen M1 regular-expression rules recovered from the outcome-blind sensitivity implementation. The labels are descriptive names used by the sensitivity script.

| Rule key | Frozen pattern | Description |
|---|---|---|
| assert | \bassert\b | assert statement |
| expected_rhs_equality | ==\s*expected | expected on right of equality |
| expected_lhs_equality | expected\s*== | expected on left of equality |
| printed_equality | print\(.*==.*\) | printed equality comparison |
| check_definition | defcheck\( | check-function definition |
| test_definition | deftest_ | test-function definition |
| unittest | \bunittest\b | unittest reference |
| pytest | \bpytest\b | pytest reference |
| validate_call | validate\( | validate-function call |
| verify_call | verify\( | verify-function call |
| length_equality | len\(.*\)\s*== | length equality |
| sum_equality | sum\(.*\)\s*== | sum equality |
| sum_upper_bound | sum\(.*\)\s*<= | sum upper-bound check |
| check_capacity | check_capacity | task-specific capacity check identifier |
| check_valid | check_valid | task-specific validity check identifier |

The leave-one-rule-out analyses retain the same blocked Poisson RELIABILITY-versus-SPEED contrast within model-family×task×wording blocks. All 15/15 derived endpoints remain positive; the most attenuated retains 80.18% of the original mean difference. The pre-specified M1_STRICT_SENSITIVITY subset contains assert, both expected-equality directions, printed equality, def check(, def test_, unittest, and pytest. It deliberately excludes generic validate(/verify( calls and task-specific identifiers, and retains 94.60% of the original effect. These analyses establish robustness to the frozen lexical rule choice; they do not convert M1 into a semantic or latent-state measure.

A separate prospective replication uses 12 externally sourced BigCodeBench tasks (Zhuo et al., 2025), two served model families, SPEED and RELIABILITY framing, and three repetitions. Of 144 planned trajectories, 142 are authoritative-valid; the inferential unit is the task rather than the individual trajectory. The task-equal RELIABILITY-minus-SPEED M1 contrast is $+0.4083$ with 95% CI $[0.2561, 0.5606]$ and two-sided $p = 0.0001025$ ($df = 11$). This supports the directional generality of the R3 framing effect, not the separate stage-separated opportunity effect.

R3's M1 endpoint is a frozen rule-based operational measure. A prior 180-case event-selection quality check is noncomparative and is not treated as a human inter-rater reliability study; the human-coding reliability results from V2.3/RH6 therefore do not transfer to R3.

### E.2 Matched operationalization analysis

The objective-by-verification-cost design uses six tasks and two configurations. Two RELIABILITY/HIGH-COST no-decision records are excluded under the terminal-availability rule, leaving 238 of 240 trajectories. Earlier response means at least one M1 event; terminal response means explicit VERIFY. The matched RELIABILITY-minus-SPEED estimates are $+27.4435$ $[21.8116, 33.3051]$ and $+6.0593$ $[-2.2701, 13.8678]$ points. Their direct difference is $+21.3842$ $[11.6667, 31.1415]$.

The documented bootstrap uses 20,000 percentile resamples of matched model–task–repeat blocks, keeping objective-by-cost cells and both outcomes together, with seed 20260905. This preserves the paired structure. The endpoints and terminal-decision rule were specified in the study record; we treat the direct comparison as a matched analysis rather than as a separately preregistered replication.

The estimand is conditional on the available terminal decisions and jointly changes window, facet, and recording form. It is neither a timing-only effect nor validation that one endpoint is a better measure of a shared latent construct. R3 and C2 are kept within the objective-framing evidence family rather than added as unrelated replication sample sizes.

# F Replay Experiments

## F.1 Quota display and an isolated revealing replay

MR1 uses 384 assignments across eight tasks, three configurations, four replay conditions, and four repeats. Each condition contains one replayed evaluation; displayed quota is one or three, and the replay is nonrevealing or defect-revealing. Correction counts are 28 resolved plus one unclassifiable in Q1/nonrevealing, 24 in Q3/nonrevealing, 27 in Q1/revealing, and 24 in Q3/revealing, each out of 96 assigned records. The unresolved semantic value is retained as missing; it is not silently coded zero.

Enumerating the two possible values of the single unresolved response gives a quota contrast of about $-4.17$ to $-3.65$ points with conservative interval union $[-9.14, 1.21]$, a revealing-content contrast of about $-1.04$ to $-.52$ with union $[-6.02, 4.34]$, and an interaction of about $+1.04$ to $+2.08$ $[-8.68, 12.04]$. These simple quota/replay manipulations therefore do not reproduce the original large positive first-action shift. We use this result only to rule against a simple quota-display account in this replay setting.

## F.2 Redundant repetition

MR2 fixes displayed quota at three and repeats the same unique nonrevealing evaluation content one, two, or three times. There are 288 planned and observed assignments: correction counts 27/96, 11/96, and 9/96. The primary two-minus-one effect is $-16.667$ points $[-23.739, -9.594]$ with 100,000-draw two-sided randomization $p \approx .000050$. Three-minus-two is $-2.083$ $[-9.35, 5.18]$; three-minus-one is $-18.75$ $[-25.42, -12.08]$; the reported saturation contrast is $-14.58$ $[-27.28, -1.89]$.

The primary direction rejects a positive one-to-two repeated-exposure account in this replay setting. It is not a general claim that repetition reduces correction: repetition also changes visible history length and the effect varies across configurations. Independent human semantic validation is not available for this replay series, so we use it as a diagnostic against a simple repetition explanation rather than as a new behavioral endpoint.

## F.3 Substantive second-step content

This replay compares a one-step baseline, a duplicated second interaction, and the historically authentic second interaction. Correction-first rates are 9.483%, 10.345%, and 60.870%, respectively. Authentic-minus-duplicate is $+50.435$ points $[40.344, 60.525]$; authentic-minus-one-step is $+51.304$ $[41.511, 61.098]$; duplicate-minus-one-step is $+.862$ $[-4.226, 5.951]$. Reported randomization $p$ values are approximately $10^{-5}$, $10^{-5}$, and 1. Thus substantive second-step content, rather than merely adding a second interaction, is sufficient to reproduce a large shift in this replay setting. The collection record does not support a clean exact-$N$ reconstruction, and independent human semantic validation remains pending, so we do not use this replay as a standalone headline result.

## F.4 Information-matched acquisition and presentation

MR4 locks 116 source histories with a candidate, task, configuration, first interaction $A_1, O_1$, and authentic second interaction $A_2, O_2$. BASE supplies the first interaction. AUTH_SELF supplies both interactions in their recorded assistant/tool roles. PASSIVE_SEQ supplies the same substantive second-step content as sequential neutral diagnostic information; PASSIVE_BUNDLE gives the same content in a neutral bundle. The recorded design holds the candidate, common instructions, tools, configuration, sampling settings, and Stage 2 allowance fixed while varying presentation. This is a transcript intervention, not evidence that the current continuation internally remembers having acquired that information itself.

Four presentations per source give 464 canonical assignment slots with randomized order (seed 2026091604). There are 456 observed responses and eight infrastructure-missing records, four in each passive condition. The report's eight "semantic unclassifiable" entries are those same eight missing records, not eight additional observed semantic ambiguities.

Table 11: MR4 frozen native-tool-proxy composition, before the content sensitivity. Rates use observed denominators; paired effects use complete-block denominators.

| Condition | Correct.-first | Eval. | No action | Other | Missing | Observed rate |
|---|---|---|---|---|---|---|
| BASE | 12 | 104 | 0 | 0 | 0 | 10.345% |
| AUTH_SELF | 71 | 40 | 5 | 0 | 0 | 61.207% |
| PASSIVE_SEQ | 59 | 46 | 7 | 0 | 4 | 52.679% |
| PASSIVE_BUNDLE | 58 | 48 | 5 | 1 | 4 | 51.786% |

The frozen primary contrast is AUTH_SELF minus PASSIVE_SEQ on complete source pairs. Reported intervals use the sample standard deviation of paired differences divided by $\sqrt{n}$ and a 1.96 multiplier; randomization uses 100,000 paired sign flips with plus-one correction. The analysis implementation was finalized after collection using the documented endpoints, contrasts, seeds, and decision rule; we therefore treat these replay results as supporting analyses rather than a fully pre-implemented confirmatory pipeline.

Table 12: MR4 frozen native-tool-proxy paired effects; estimates and intervals are percentage points.

| Contrast | Complete pairs | Effect | 95% CI |
|---|---|---|---|
| AUTH_SELF − BASE | 116 | 50.8621 | $[41.1149, 60.6092]$ |
| PASSIVE_SEQ − BASE | 112 | 41.9643 | $[32.1333, 51.7953]$ |
| PASSIVE_BUNDLE − BASE | 112 | 41.0714 | $[30.9569, 51.1860]$ |
| AUTH_SELF − PASSIVE_SEQ | 112 | 8.9286 | $[-.2237, 18.0808]$ |
| AUTH_SELF − PASSIVE_BUNDLE | 112 | 8.9286 | $[-.2237, 18.0808]$ |
| PASSIVE_SEQ − PASSIVE_BUNDLE | 111 | 0.0000 | $[-9.7154, 9.7154]$ |

The authentic-versus-BASE comparison reproduces a large shift. For the information-matched contrast, AUTH_SELF minus PASSIVE_SEQ is $+8.93$ points $[-0.22, 18.08]$ with randomization $p = .0874$. The interval spans both zero and values above a $+10$-point practical margin, so this comparison neither establishes equivalence nor a practically large residual effect of acquisition form.

**Semantic sensitivity.** The frozen proxy classifies native `write_file` calls as correction and native `execute_code` calls as evaluation. Inspection of the canonical first-action records finds 37 `execute_code` calls that contain literal module-level writes to `candidate.py`, plus one malformed correction-oriented attempt. We therefore report two post-hoc semantic sensitivities without replacing the frozen endpoint. Reclassifying the 37 literal writes reduces AUTH_SELF minus PASSIVE_SEQ to $+2.68$ points $[-4.97, 10.32]$; also including the malformed attempt gives $+3.57$ $[-3.86, 11.00]$. These sensitivities are consistent with a substantial role for shared diagnostic content, but they are not exhaustive semantic recodings or independent human validation.

Table 13: Post-hoc semantic sensitivity for the information-matched replay. Effects are authentic minus passive sequential on the same 112 complete pairs; the frozen proxy result is shown for reference.

| Endpoint | Effect (pp) | 95% interval |
|---|---|---|
| Frozen native-tool proxy | 8.9286 | $[-.2237, 18.0808]$ |
| Plus 37 literal module-level candidate writes | 2.6786 | $[-4.9676, 10.3247]$ |
| Also include one malformed correction attempt | 3.5714 | $[-3.8569, 10.9998]$ |

Under the literal-write sensitivity, BASE remains 10.345% correction-first, while AUTH_SELF, PASSIVE_SEQ, and PASSIVE_BUNDLE become 68.966%, 65.179%, and 64.286%. The replay series therefore narrows the interpretation toward substantive diagnostic content while leaving residual acquisition-form effects uncertain. We use these analyses as supporting interpretation, not as a causal mediation decomposition or an additional headline contribution.

Table 14: Stage 1 accepted-execution diagnostics. Within-HIGH correction-first comparisons are descriptive.

| Cohort | Arm | Mean accepted | $n$ at two | $n$ at three | Correction-first: two / three |
|---|---|---|---|---|---|
| V2.3 | LOW | 1.0000 | 0 | 0 | not applicable |
| | HIGH | 2.8403 | 23 | 121 | 73.91% / 74.38% |
| RH6 | LOW | 1.0000 | 0 | 0 | not applicable |
| | HIGH | 2.7708 | 22 | 74 | 68.18% / 71.62% |

## G Realized Evaluation and Conditional Compensation

Accepted-execution counts provide a manipulation diagnostic: LOW uses exactly one accepted evaluation in both cohorts, whereas HIGH averages 2.84 in V2.3 and 2.77 in RH6. Within-HIGH correction-first rates are similar for records with two versus three accepted evaluations, but this descriptive comparison does not identify a dose-response curve because realized counts are post-treatment and LOW never enters those count levels.

A separate descriptive summary counts recorded evaluation actions before the first correction attempt. Its denominators refer to trajectories reaching that action, not to verified successful repair. Because later evaluation-count semantics are not identical to the accepted Stage 1 credit counter, the total is treated as a descriptive trajectory count rather than a uniform compute or information dose.




Figure 5: Reported conditional compensation: LOW performs more later checking, but fewer total recorded evaluations precede its correction attempt. This display does not identify a mediator, compare equal selected populations, or measure successful repairs. See the extraction and counting qualifications above.

Later-action coding contains ambiguous cases, and documented extreme recodings preserve the qualitative direction of the descriptive compensation pattern. These summaries remain secondary anatomy: they do not turn realized action counts into a randomized mediator or imply equal total computation across arms.

## H Reproducibility Resources and Scope

The supporting-materials release contains frozen assignment-level outputs and analysis code for the prospective V2.3 and RH6 cohorts; the R3 assignment manifest, authoritative ledger, 1,404-row analysis dataset, and confirmatory analysis code; the C2 matched dataset and analysis implementation; the complete M1 rule-sensitivity specification, results, and tests; and the external BigCodeBench replication package. It also includes the prospective human-coding labels, adjudication records, and coding locks, together with the recovered files used for the historical same-object analysis.

These artifacts are sufficient to regenerate the reported numerical summaries from the frozen analysis data and to inspect the main coding and recovery decisions. They are not a complete archive of every provider-side interaction or sandbox state from every historical branch, and the later replay series is less complete than the prospective cohorts. We therefore treat replay as supporting interpretation and do not claim end-to-end rerunnability of every historical experiment from the released bundle alone.