COVERAGE IS NOT POLICY VALUE: AN AUDIT OF TEST-TIME COMPUTE ORACLES

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time-compute studies often report an oracle that selects whichever stochastic action eventually succeeds on each problem. This measures retrospective coverage, which need not equal the value of choosing an action before its outcome is known. We distinguish retrospective coverage, the expected-action oracle (the best population action mean), finite-replication split-selected value, and held-out policy value. We show that coverage can grow with the number of exchangeable actions even when no decision-time policy improves over a fixed action. Our evaluation uses 771 model-problem states from three open models on MATH-500 and GPQA-Diamond, with five prompt actions and 16 outcomes per stochastic action. Under empirical product coupling, retrospective coverage exceeds split-selected value by 4.41 percentage points on MATH (95% problem-bootstrap interval [3.63, 5.26]) and 11.73 points on GPQA ([10.34, 13.17]); all six model-benchmark cells have positive gaps. Although split selection improves over a fixed prompt action, four router families evaluated on held-out problems show no statistically supported gain in either the prompt-action bank or a second bank that treats models as actions. Oracle headroom should therefore be reported with its outcome coupling and evaluated separately from decision-time policy value.

1 INTRODUCTION

Increasing inference-time computation can improve language-model reasoning by sampling more candidates, allocating different budgets, or changing the reasoning protocol (Wang et al., 2023; Snell et al., 2025; Damani et al., 2025; Zuo & Zhu, 2026). Before building an adaptive policy, studies often measure an oracle: for each problem, select whichever candidate, budget, or action eventually succeeds. This answers an important coverage question. It does not necessarily answer whether a decision rule can identify the useful action from information available before that action is executed.

The distinction is consequential for stochastic actions. Suppose five actions are exchangeable and each succeeds with probability p. No action is more valuable than another before execution, yet the probability that at least one succeeds is $1 - (1 - p)^5$, which can be much larger than p. If action means are estimated from a few outcomes, maximizing those estimates adds a second source of optimism. A large realized oracle gap may therefore combine coverage, finite-sample maximization, and actionable heterogeneity without identifying how much each contributes.

We study the statistical interface between replicated action outcomes and claims about test-time-compute headroom. The central object is a model-problem state followed by a choice among stochastic actions. We distinguish the retrospective coverage oracle (RCO), the expected-action oracle (EAO), the value of a finite-replication selection rule, a globally fixed action, and a deployable policy. The distinctions concern available information rather than terminology: RCO observes all realized potential outcomes, whereas a policy must act before observing any of them.

RQ1. Which quantities do replicated stochastic action banks identify when action outcomes are observed before or after selection?

RQ2. How much product-coupled retrospective coverage remains after independent split-panel selection and evaluation?

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

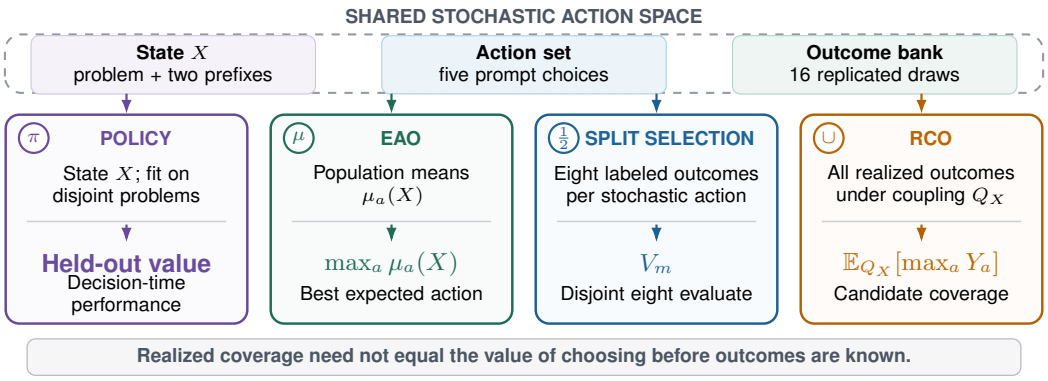


Figure 1: The same problem and action space give different values according to the information available when choosing. A policy uses only X and is evaluated on held-out problems. EAO assumes knowledge of population action means; split-selected value uses independent selection and evaluation panels; and RCO uses realized outcomes under a declared coupling.

RQ3. How much of this headroom is retained by policies that use only decision-time information on held-out problems?

Contributions. First, we define retrospective coverage, expected-action value, split-selected value, and deployable policy value on one stochastic action space. The accompanying results characterize coupling dependence, plug-in optimism, and finite-panel evaluation. Second, we provide a symmetrized selection/evaluation procedure, with exact enumeration for common panels and seeded Monte Carlo for independently split action panels, alongside problem-level inference and action-label stability diagnostics. Third, a generation protocol frozen before evaluation audits five prompt actions across three open models and two reasoning benchmarks. Fourth, post-outcome analyses, whose protocols and design status are reported explicitly, use held-out structured and semantic routers to test whether the observed headroom is predictable. A second action bank treats the three models as actions and exhibits the same separation between label-informed selection and the tested decision-time policies.

2 Related Work

Test-time scaling and adaptive computation. Repeated sampling and self-consistency use additional inference to improve coverage or aggregate agreement (Wang et al., 2023; Brown et al., 2024; Schaeffer et al., 2025). Compute-optimal inference instead allocates search or sampling according to problem difficulty (Snell et al., 2025; Damani et al., 2025). Early-stopping methods use agreement or answer-discovery processes to avoid unnecessary samples (Aggarwal et al., 2023; Li et al., 2024). These methods optimize how compute is spent. Our analysis addresses the preceding measurement question: what a replicated action matrix establishes about coverage, selection, and routing.

Bandit formulations allocate generation budget across queries and distinguish response coverage from accuracy (Zuo & Zhu, 2026). Our setting instead holds the evaluation population fixed and studies the choice among stochastic protocols or models within a problem. Replicated outcomes expose how coupling, finite-panel selection, and decision-time prediction change the value assigned to that choice.

Candidate coverage and selection. Pass@k and candidate-pool oracles quantify whether a successful response is present, a useful upper bound for reranking or verification (Chen et al., 2021; Brown et al., 2024). Imperfect verifiers can prevent that coverage from becoming realized accuracy (Stroebl et al., 2026). Difficulty-oracle experiments likewise provide diagnostic upper bounds, while learned policies are evaluated separately (Snell et al., 2025). Our framework preserves this legitimate coverage interpretation and separates it from expected action value. The second issue is post-selection reuse: the maximum of noisy estimates is optimistic even when each estimate is unbiased (Andrews

et al., 2024). Our focus is per-state stochastic actions, for which both action identity and utility may be uncertain and a one-hot oracle label may be unstable.

Closest evaluation and routing studies. Recent work learns constrained per-input compute policies (Zhai et al., 2026), diagnoses recoverable mass in fixed candidate pools (Hu, 2026), and separates end-to-end systems from candidate-bank diagnostics (Hariri et al., 2026). Cost-aware protocol routing explicitly reports a single-realization oracle as a diagnostic rather than expected per-instance utility (Yang et al., 2026). SIREN addresses winner bias when a global prompt or program is selected through adaptive benchmark reuse (Xu et al., 2026). We instead study a different sampling axis: each problem has multiple stochastic actions, each action has replicated outcomes, and the target is the value of selecting an action for that problem from a finite independent panel.

Evaluation methodology. Benchmark variance can affect the interpretation of model differences (Madaan et al., 2024), while test-time scaling comparisons require a specified inference protocol (Hariri et al., 2026). Our procedure applies these concerns to an action tensor: the problem is the resampling unit, selection outcomes are disjoint from evaluation outcomes, and all failures remain in the denominator. Unlike a global prompt-selection analysis, the selected action can differ by problem; unlike a policy evaluation, its selection panel contains labeled potential outcomes and is therefore a measurement device rather than a deployable input.

3 STOCHASTIC ACTION ESTIMANDS

Let X contain exactly the information available before an action is chosen, and let $\mathcal{A} = \{1, \dots, A\}$ be the action set. Action a has bounded potential utility $Y_a \in [0, 1]$ and conditional mean $\mu_a(x) = \mathbb{E}[Y_a \mid X = x]$. A coupling Q_x specifies the joint law of $(Y_1, \dots, Y_A)$; the marginal action means do not uniquely determine this law.

Retrospective coverage. The RCO observes all realized action outcomes and records the largest utility:

$$V_{\mathrm{RCO}}(x; Q_x) = \mathbb{E}_{Q_x}[\max_a Y_a \mid X = x]. \tag{1}$$

This quantity is a property of both the action marginals and their coupling. For binary outcomes it is the probability that at least one action succeeds, so it is appropriate when the question is whether a successful candidate exists.

Expected-action oracle. An oracle that knows conditional action means but must choose before execution has value

$$V_{\mathrm{EAO}}(x) = \max_a \mu_a(x). \tag{2}$$

EAO is a population quantity. With m outcomes per action, the observable plug-in maximum $\max_a \widehat{\mu}_a$ is an estimator, not EAO itself.

Split-selected value. Let S and E be independent panels of m outcomes per action. Selection weights $w_a(S)$ are uniform over actions tied for the largest selection-panel mean. The selected rule has value

$$V_m(x) = \mathbb{E}_S \left[\sum_a w_a(S) \mu_a(x) \right], \tag{3}$$

estimated by $\sum_a w_a(S) \bar{Y}_a^E$. This rule is informative about finite-replication identifiability, but its labeled selection outcomes are not available to a deployed router.

Fixed and deployable values. A fixed action is selected using disjoint fitting problems and then evaluated on all evaluation problems. A deployable policy $\pi(a \mid X)$ has value $\mathbb{E}_X \sum_a \pi(a \mid X) \mu_a(X)$ and may use only X. These references answer different questions: the fixed action measures global action quality; the split-selected rule measures label-informed per-state selection; and the policy measures achievable decision-time value.

Table 1: The same replicated action bank supports distinct scientific questions. "Outcome access" records which potential outcomes inform the choice, not which data are later used to estimate its value.

Quantity	Outcome access when choosing	Target	Appropriate interpretation
Fixed action	None at the evaluation state	Value of one action selected on disjoint fitting problems	Global action-quality reference
Split-selected value V_m	Labeled selection panel from the same state	Expected utility of the induced empirical-maximizer rule	Replication-limited identifiability, not deployment
Expected-action oracle (EAO)	Population conditional means	$\max_a \mu_a(X)$	Ideal one-action choice before execution
Retrospective coverage oracle (RCO)	All realized potential outcomes	$\mathbb{E}[\max_a Y_a \mid X]$ under coupling Q_X	Candidate coverage after realizations are known
Held-out policy	Only observable state features X	Held-out value of $\pi(a \mid X)$	Achievable decision-time routing

4 What finite action matrices identify

Proposition 1 (Coverage without routability)**.** *If all A actions have conditional success probability $p(x)$, then $V_{\mathrm{EAO}}(x) = p(x)$. Under the product coupling, $V_{\mathrm{RCO}}(x) = 1 - (1 - p(x))^A$. Every policy measurable with respect to X still has conditional value $p(x)$.*

The proof follows by taking the complement of joint failure. Thus RCO can grow with the number of exchangeable actions even when there is no decision-time signal with which to route among them.

Proposition 2 (Coupling bounds)**.** *For fixed distributions on $[0, 1]$,*

$$\max_a \mu_a(x) \le V_{\mathrm{RCO}}(x; Q_x) \le \min\left\{1, \sum_a \mu_a(x)\right\}. \tag{4}$$

Both bounds are sharp over Bernoulli marginals. For independent Bernoulli outcomes, $V_{\mathrm{RCO}} = 1 - \prod_a (1 - \mu_a)$.

Nested Bernoulli success events attain the lower bound; disjoint events attain the upper bound until their total probability covers the unit interval. The bounds in equation 4 make coupling part of an RCO specification rather than an implementation detail.

Proposition 3 (Plug-in optimism)**.** *For unbiased action-mean estimators, $\mathbb{E}[\max_a \widehat{\mu}_a] \ge \max_a \mu_a$. If each mean uses m independent outcomes in $[0, 1]$, then*

$$0 \le \mathbb{E}[\max_a \widehat{\mu}_a] - \max_a \mu_a \le \sqrt{\frac{\log A}{2m}}. \tag{5}$$

Convexity gives the lower inequality. Hoeffding's lemma and a log-sum-exp bound give the upper inequality. The bound describes a worst case; optimism can be zero for a deterministic dominant action.

Proposition 4 (Target of split-panel evaluation)**.** *If E is conditionally independent of S given X, then*

$$\mathbb{E}\left[\sum_a w_a(S) \bar{Y}_a^E \mid S, X\right] = \sum_a w_a(S) \mu_a(X). \tag{6}$$

Hence split-panel evaluation is unbiased for the split-selected rule averaged over S, but not generally for EAO.

Proposition 5 (No finite-sample unbiased EAO estimator)**.** *For any fixed finite number of Bernoulli observations, no finite-valued statistic is unbiased for EAO over the unrestricted action-mean space when $A \ge 2$.*

The expectation of every finite-sample statistic is a polynomial in the action probabilities, whereas $\max(\mu_1, \mu_2)$ is not a polynomial on the unit square. This does not preclude consistent estimation, useful shrinkage, or valid bounds. Full proofs and the polynomial argument are in Appendix A.

5 REPLICATION-AWARE ORACLE EVALUATION

For each state, we retain two immutable initial responses and evaluate five actions: stop, retry the same prompt, request a different approach using unverified answer-mode counts, critique an untrusted representative trace, and decompose the problem without history. Stop chooses uniformly among plurality modes in the two-response prefix; its accuracy is evaluated fractionally. Every stochastic action has 16 fresh outcomes.

The plug-in maximum uses all 16 outcomes to estimate each action mean and takes their per-state maximum. For split-panel selection, eight outcomes per action select the empirical maximizer and the complementary eight evaluate it. Since independently generated actions have no intrinsic cross-action ordering, the primary analysis uses independent partitions per stochastic action. It averages 500,000 seeded joint partitions across five runs. RCO is the exact expectation under independent action permutations, which is the empirical product coupling. As a sensitivity, we also use common complementary panels and align outcomes by their arbitrary replicate indices; this version averages exactly over all $\binom{16}{8} = 12{,}870$ partitions.

Uncertainty intervals resample problems. In benchmark-pooled analyses, a resampled problem retains its rows from all three models. We report tie-aware agreement between selected-action distributions from disjoint 1/1, 2/2, 4/4, and 8/8 panels. Jeffreys posteriors over stochastic action means provide action-optimality probabilities; stop retains its deterministic fractional value. Simultaneous marginal action comparisons use two-sided paired randomization tests with Holm correction within each model-benchmark cell. Unresolved parses, exceptions, and truncated generations remain in the full denominator.

Held-out policies. Four router families use only information available before the selected action is executed. Structured and signed-hash ridge models use problem metadata, length descriptors, and, for prompt routing, summaries of the two-response prefix. A frozen Qwen3-Embedding-0.6B encoder supplies semantic features to a ridge head and a cosine nearest-neighbor rule. Hyperparameters are selected by five-fold cross-validation on fitting problems. Evaluation problems are not used for feature design, fitting, tuning, or tie-breaking. All policy comparisons are paired against the fixed action chosen on the same fitting population.

6 EMPIRICAL AUDIT

Benchmarks and splits. MATH-500 contains competition mathematics problems with equivalence-based answer scoring (Lightman et al., 2024; Hendrycks et al., 2021). Fixed actions and routers use 250 fitting problems for Qwen3-8B and R1-Distill-7B, and 50 for Phi-3.5-mini. A disjoint set of 125 problems per model forms evaluation. GPQA-Diamond contains expert-written, four-option graduate science questions (Rein et al., 2024); 66 problems select global fixed actions and 132 disjoint problems form evaluation. Answer choices are deterministically permuted once per problem. A constrained one-token readout returns an option letter after the complete reasoning response and does not receive the reference answer.

Models and generation. We evaluate Qwen3-8B in thinking mode (Yang et al., 2025), DeepSeek-R1-Distill-Qwen-7B (Guo et al., 2025), and Phi-3.5-mini-instruct (Microsoft, 2024). All checkpoints and dataset snapshots are pinned to exact revisions. Stochastic generations use temperature 0.6, top-p 0.95, and at most 24,576 completion tokens. Prompts never include correctness labels or reference answers. The global fixed action for each model-benchmark cell is chosen only on disjoint fitting problems.

Scale and integrity. The evaluation has 771 model-problem states and 50,886 stochastic outcomes; the corresponding 248 fitting states add 16,368 outcomes, for 67,254 generated outcomes in this audit. Every attempt, prompt, state, configuration, source, and output shard is hashed. Evaluation matrices have complete action-replicate coverage and unique identifiers. GPQA has 100% parse coverage. On MATH, parse rates across model-action cells range from 98.15% to 99.80%; failed parses remain incorrect. For tabular analysis, representing each fractional stop value across 16 replicate slots yields 61,680 action-replicate rows.

Table 2: Information and computation used by the five prompt actions. Prefix responses are unverified. No action receives a reference answer or correctness label.

Action	Information supplied	Operation	New continuation
Stop	Two prefix answer modes	Choose uniformly among plurality modes	No
Retry	Original problem	Sample the unchanged problem prompt	Yes
Diversify	Problem and prefix answer-mode counts	Seek an independent approach	Yes
Critique	Problem, answer modes, and one untrusted prefix trace	Locate an error, then solve independently	Yes
Decompose	Original problem	Form subgoals, solve, and verify	Yes

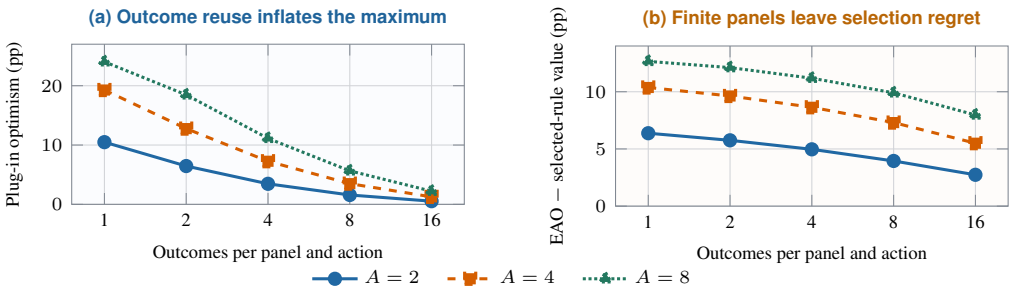


Figure 2: Finite-replication simulation with 500 problems and 2,000 datasets per condition. Each problem has one action with success probability 0.75 and all others 0.60. Reusing outcomes in the plug-in maximum is optimistic (left); independent evaluation targets the split-selected rule, whose value approaches but need not equal EAO at finite replication (right).

Action banks and primary quantities. The prompt-action bank contains stop, retry, diversification, critique, and decomposition. Table 2 states the information used by each action. A second bank uses the three models as actions and evaluates a fresh retry outcome. It uses the 50 MATH and 66 GPQA fitting problems common to all models, followed by the common evaluation populations. For each bank we report the fit-selected fixed action, held-out policy value, split-selected value V_m, plug-in maximum, and product-coupled RCO. Percentile intervals use 10,000 problem-bootstrap replicates. The prompt generation and initial matched-panel analysis protocol were frozen before evaluation generation. The product-coupling, policy, and model-routing analyses were specified after the outcome banks existed and are reported as post-outcome analyses. Product coupling is the primary RCO specification because independently generated actions have no intrinsic cross-action replicate alignment.

7 Results

Simulation separates maximization bias from selection regret. We simulate 270 conditions that vary the number of actions, outcomes per panel, problems, base success rate, and action separation, with 2,000 datasets per condition. Split-panel evaluation differs from its target by at most 0.13 percentage points across the grid. In contrast, plug-in optimism for EAO is nonnegative in every condition and reaches 44.79 points. Figure 2 shows that additional actions amplify maximization bias and make the best action harder to identify, while replication reduces both effects.

Coverage exceeds split-selected value. Table 3 shows a positive RCO-minus-V_m gap in all six model-benchmark cells under empirical product coupling. The pooled gap is 4.41 percentage points on MATH and 11.73 points on GPQA. Their 95% intervals are [3.63, 5.26] and [10.34, 13.17]. Split-panel selection improves over the fit-selected fixed action by 3.99 points on MATH and 13.36 points on GPQA. Nevertheless, 52.5% and 46.7% of product-coupled RCO headroom, respectively, is

Table 3: Full-denominator results under empirical product coupling. Split-selected value uses independent 8/8 panels per stochastic action. Intervals resample problems 10,000 times. "Not retained" is $(\mathrm{RCO} - V_m)/(\mathrm{RCO} - \mathrm{fixed})$.

Model	Benchmark	Fixed	Plug-in	V_m	RCO	RCO$-V_m$ (95% CI)	Not retained
Qwen3-8B	MATH	95.85	98.05	97.51	98.69	1.19 [0.45, 2.14]	41.8
R1-Distill-7B	MATH	94.15	97.15	96.22	97.74	1.51 [0.74, 2.54]	42.2
Phi-3.5-mini	MATH	44.85	56.80	53.08	63.61	10.53 [8.37, 12.79]	56.1
Qwen3-8B	GPQA	53.12	69.08	66.24	74.52	8.28 [6.33, 10.41]	38.7
R1-Distill-7B	GPQA	25.90	38.54	35.56	43.73	8.17 [5.86, 10.67]	45.8
Phi-3.5-mini	GPQA	30.30	52.98	47.60	66.34	18.74 [16.14, 21.33]	52.0
Pooled	MATH	78.28	84.00	82.27	86.68	4.41 [3.63, 5.26]	52.5
Pooled	GPQA	36.44	53.54	49.80	61.53	11.73 [10.34, 13.17]	46.7

Table 4: Information ladder for two action banks (accuracy in percent). The held-out column gives the range of paired differences from the fit-selected fixed action across structured, hashed-text, semantic-ridge, and semantic-kNN routers. No router is selected using evaluation outcomes.

Action bank	Benchmark	Fixed	Held-out policy − fixed	V_m	Plug-in	Product RCO
Prompt	MATH	78.28	[-0.75, +0.40]	82.27	84.00	86.68
Prompt	GPQA	36.44	[-3.12, -0.25]	49.80	53.54	61.53
Model	MATH	95.45	[-1.20, +0.00]	96.08	96.90	97.33
Model	GPQA	46.31	[-9.19, -5.30]	60.06	62.64	68.53

not retained. Thus replicated labels reveal useful per-state differences, but realized coverage remains substantially larger.

The tested held-out policies do not retain this headroom. Table 4 compares four router families without selecting among them on evaluation outcomes. For the prompt bank, pooled differences from the fixed action range from -0.75 to $+0.40$ points on MATH and from -3.12 to -0.25 points on GPQA. Every MATH interval includes zero. The semantic-ridge difference on GPQA is -3.12 points, with a 95% interval of [-5.38, -0.96]. The other families also provide no supported gain.

The model-action bank strengthens the separation. On GPQA, the fixed Qwen3-8B action attains 46.31%, while split-panel selection reaches 60.06%, the plug-in maximum reaches 62.64%, and product RCO reaches 68.53%. In contrast, the four held-out policies fall between 5.30 and 9.19 points below the fixed action. On MATH, the fixed model already attains 95.45%; the held-out policy differences range from -1.20 to 0.00 points. These results do not exclude a different router or a larger fitting population. They show that the observed selectable and retrospective headroom is not automatically available to decision-time policies.

Hard action labels remain unstable. Agreement rises with replication but remains 45.0% on GPQA and 26.9% on MATH at 8/8 (Figure 3). The MATH result partly reflects a ceiling: several actions have similar utility for the two reasoners, so labels change even when their utilities are close. The fraction of states whose largest posterior action-optimality probability is below 0.8 is 18.7% on MATH and 51.5% on GPQA. Hard labels therefore discard ties and graded uncertainty, which helps explain why routers trained from small panels can fail to generalize.

Declared analysis variants preserve the prompt-bank separation. With matched replicate indices and common complementary panels, the pooled RCO-minus-V_m gaps are 4.48 points on MATH and 11.87 on GPQA, which differ from the product-coupled values by only 0.07 and 0.14 points. Under that matched sensitivity, deterministic tie resolution gives gaps of 3.28 and 9.27 points, while removing stop gives 4.63 and 14.09 points. Completion-token normalization has positive intervals, and parsed-only evaluation preserves the direction in all six cells. Appendix D reports each cell and the associated denominator convention.

Coupling changes the meaning of coverage. In Figure 4, positive alignment leaves every action mean fixed but contracts RCO minus EAO. With four actions, the gap falls from 23.40 points under independence to 2.34 points at coupling 0.9. The small empirical difference between product and

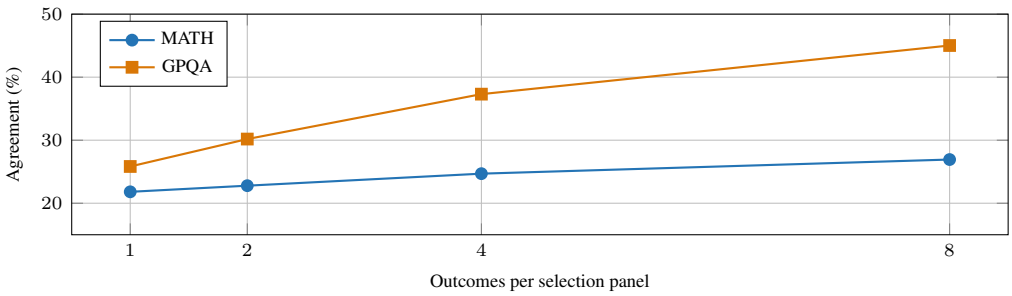


Figure 3: Tie-aware agreement between best-action distributions from two disjoint panels. Agreement rises with replication but remains 26.9% on MATH and 45.0% on GPQA at 8/8. Ties receive fractional credit.

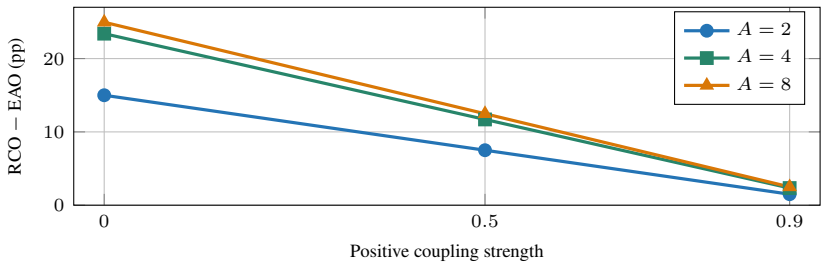


Figure 4: Controlled coupling simulation for 500 problems, eight outcomes per selection and evaluation panel, base success 0.60, and a 0.15 best-action advantage. Aligning action successes contracts coverage over EAO even though every marginal action mean is unchanged.

matched constructions is therefore a property of this action bank, rather than a reason to omit the coupling from the RCO definition.

8 Implications for Oracle Reporting

The contraction has two conceptually distinct components. Averaging over states gives the exact decomposition

$$\mathbb{E}[V_{\mathrm{RCO}}(X) - V_m(X)] = \mathbb{E}[V_{\mathrm{RCO}}(X) - V_{\mathrm{EAO}}(X)] + \mathbb{E}[V_{\mathrm{EAO}}(X) - V_m(X)]. \tag{7}$$

The first term is realized coverage beyond the best action mean and depends on cross-action coupling. The second is the regret of selecting an action from a finite replicated panel. Both are nonnegative: pointwise RCO dominates every action realization in expectation, and EAO is the largest conditional action mean. Our split-panel estimator targets the left-hand side for the specified finite selection rule. Separating its two components requires EAO, for which no unbiased finite-sample estimator exists over the unrestricted action-mean space. Substituting the plug-in maximum would instead reintroduce maximization bias.

An oracle number should be accompanied by the information required to identify its scientific meaning. For stochastic action banks, this includes the action set, outcome coupling, replications per action, whether selection outcomes are reused for evaluation, tie handling, problem-level uncertainty, and the information available at decision time. Coverage is most informative when the system can generate or verify several candidates; expected-action value is appropriate when exactly one action must be selected; and policy value requires a decision rule evaluated on held-out problems.

Best-action labels deserve the same scrutiny as scalar oracle values. A one-hot label treats a narrow empirical win as certain and makes tied actions depend on an arbitrary convention. Replicated utility vectors, posterior optimality probabilities, or uncertainty-aware best-action sets preserve more of what the action matrix establishes. This matters for supervised routing: fitting to a noisy argmax can target replication noise rather than a stable decision.

The policy results expose a further distinction. Both prompt-action and model-action banks contain label-informed headroom, yet none of the four tested router families improves reliably over its fit-selected fixed action on held-out problems. This contrast is evidence about these policies under the observed fitting populations, not an impossibility result. It nevertheless shows why a coverage or split-selected number should not be presented as an achievable policy gain without an independently evaluated router.

The estimands can be applied to other action banks without adopting the five actions used here. Such an analysis requires problem, action, replicate, outcome, cost, split, model, and benchmark identifiers, together with provenance sufficient to establish independent generation and held-out evaluation.

A minimal reporting contract. An oracle claim should state (i) what is observed before the choice, (ii) which outcomes are exposed to the oracle, (iii) the coupling used to align stochastic outcomes, (iv) how many independent outcomes estimate each action, (v) whether selection and evaluation reuse outcomes, and (vi) how ties, failures, costs, and problem-level uncertainty are handled. For a policy claim, the report must add held-out policy predictions and the features available at decision time. These fields make coverage and policy results comparable without requiring one particular action set or routing architecture.

9 LIMITATIONS

Our audit covers two reasoning benchmarks, three models no larger than eight billion parameters, and five actions. Qwen3-8B and R1-Distill-7B share a Qwen architectural lineage; Phi-3.5-mini provides a distinct family but not a large scale. The measured contraction may differ for code execution, long-horizon agents, proprietary models, learned verifiers, or action sets with stronger separation.

The split-selected rule is deliberately label-informed. It quantifies what finite replication can identify when action outcomes are available; it is not a deployable router. Establishing policy value requires observable features and held-out policy predictions. Conversely, EAO remains a population ideal rather than an unbiased finite-sample estimate. The empirical results compare RCO to a finite rule, not directly to a known EAO.

The policy audit covers four deliberately simple router families: structured ridge, signed text-hash ridge, semantic ridge, and semantic nearest neighbors. It does not cover larger supervised models, online exploration, verifier-based routing, or joint router and generator training. The fitting populations are also modest, especially the 50 MATH states shared across all three models in the model-action bank. Failure to improve over the fixed action therefore limits claims about the tested policies, rather than all possible policies.

Accuracy and completion tokens omit latency, energy, memory, and monetary cost. Fractional stop evaluation represents expected tie resolution rather than one realized decision. GPQA uses a constrained answer readout to make the multiple-choice contract consistent across models; this separates answer formatting from reasoning but adds a second deterministic model call. Finally, the PDF does not contain response-level outcome matrices or text embeddings. Reproducing raw generations requires authorized access to the benchmark sources and the stated model checkpoints.

10 CONCLUSION

Stochastic action matrices pose three questions that one oracle cannot answer. RCO measures realized coverage under a declared coupling. Split-panel evaluation targets a finite label-informed rule, while deployable value requires held-out predictions. Across six prompt-bank cells, product-coupled RCO exceeds split-selected value, although split selection improves over a fixed action. Yet neither the prompt bank nor a model-action bank yields a reliable gain for the four tested routers. Together with unstable labels and optimistic plug-in maxima, this shows that selectable or retrospective headroom need not become policy value. Declaring the estimand and coupling, then evaluating selection and policy independently, makes oracle headroom interpretable rather than an implicit policy claim.

AI USE STATEMENT

Generative AI tools were used to assist with drafting, language editing, and polishing the manuscript. The authors independently checked all experimental results, mathematical statements, citations, claims, and final text and take full responsibility for the content.

ETHICS STATEMENT

No ethics review was required because this work involved no human-subject data collection, and no human annotation enters the reported results. The study therefore imposed no task-related risk, evaluation, or employment consequence on an annotator. Any independent, voluntary annotation-like checking of the manuscript was conducted anonymously. Each annotator had complete discretion to stop or withdraw at any time for personal reasons, without penalty or adverse consequence, and withdrawal could not affect the annotator. The experiments use public or access-controlled reasoning benchmarks and open model checkpoints. No restricted GPQA questions, answer choices, reference answers, or raw generations are included in this paper; access to the underlying datasets and checkpoints remains subject to their respective terms.

REPRODUCIBILITY STATEMENT

Sections 3–7 define the estimands, assumptions, estimator, datasets, action bank, generation settings, scoring, splits, and full-denominator analysis. The appendix provides proofs, action and failure-handling definitions, generation and scoring details, statistical analyses, and held-out routing details. Independent regeneration requires access to the stated benchmarks and model checkpoints, including the restricted GPQA source.

REFERENCES

Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. Let's sample step by step: Adaptive-consistency for efficient reasoning and coding with LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12375–12396, 2023. doi: 10.18653/v1/2023.emnlp-main.761.

Isaiah Andrews, Toru Kitagawa, and Adam McCloskey. Inference on winners. *The Quarterly Journal of Economics*, 139(1):305–358, 2024. doi: 10.1093/qje/qjad043.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Mehul Damani, Idan Shenfeld, Andi Peng, Andreea Bobu, and Jacob Andreas. Learning how hard to think: Input-adaptive allocation of LM computation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ff414825df833edb8b1839e3d5d495e9-Abstract-Conference.html.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Mohsen Hariri, Weicong Chen, Nahal Shahini, Vikash Singh, Kai Ye, Amirhossein Samandar, Debargha Ganguly, Sreehari Sankar, Yanyan Zhang, Shouren Wang, Jerry Peng, Biyao Zhang, Michael Hinczewski, and Vipin Chaudhary. Test-time scaling in reasoning LLMs: Inference regimes, evaluation, and reproducibility. *arXiv preprint arXiv:2608.04001*, 2026.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://openreview.net/forum?id=7Bywt2mQsCe.

Jie Hu. Oracle gap and signal fidelity: A fixed-pool diagnostic for test-time collaboration. *arXiv preprint arXiv:2607.17531*, 2026.

Yiwei Li, Peiwen Yuan, Shaoxiong Feng, Boyuan Pan, Xinglin Wang, Bin Sun, Heda Wang, and Kan Li. Escape sky-high cost: Early-stopping self-consistency for multi-step reasoning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/3fe2a777282299ecb4f9e7ebb531f0ab-Abstract-Conference.html.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/aca97732e30bcf1303bc22ac3924fd16-Abstract-Conference.html.

Lovish Madaan, Aaditya K. Singh, Rylan Schaeffer, Andrew Poulton, Sanmi Koyejo, Pontus Stenetorp, Sharan Narang, and Dieuwke Hupkes. Quantifying variance in evaluation benchmarks. *arXiv preprint arXiv:2406.10229*, 2024. URL https://arxiv.org/abs/2406.10229.

Microsoft. Phi-3.5-mini-instruct model card. https://huggingface.co/microsoft/Phi-3.5-mini-instruct, 2024. Checkpoint revision 2fe192450127e6a83f7441aef6e3ca586c338b77.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

Rylan Schaeffer, Joshua Kazdan, John Hughes, Jordan Juravsky, Sara Price, Aengus Lynch, Erik Jones, Robert Kirk, Azalia Mirhoseini, and Sanmi Koyejo. How do large language monkeys get their power (laws)? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 53132–53176, 2025. URL https://proceedings.mlr.press/v267/schaeffer25a.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1b623663fd9b874366f3ce019fdfdd44-Abstract-Conference.html.

Benedikt Stroebl, Sayash Kapoor, and Arvind Narayanan. The limits of inference scaling through resampling. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/95591b7557edea46ae1d90b0e9e825c5-Abstract-Conference.html.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

Yang Xu, Jiefu Zhang, Haixiang Sun, Zihan Zhou, Tianyu Cao, and Vaneet Aggarwal. Towards reliable LLM evaluation: Correcting the winner's curse in adaptive benchmarking. *arXiv preprint arXiv:2605.05973*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Chih-Hsuan Yang, Jingyan Jiang, Cheng-Hau Yang, Vikram Vasudevan, Huihuo Zheng, Venkatram Vishwanath, and Rajeev Thakur. LLMs can predict failure risk, but struggle to predict which collaboration protocol pays off: Cost-aware protocol routing across reasoning tasks. *arXiv preprint arXiv:2608.14927*, 2026.

Zhiyuan Zhai, Bingcong Li, Bingnan Xiao, Ming Li, and Xin Wang. Adaptive test-time compute allocation for reasoning LLMs via constrained policy optimization. *arXiv preprint arXiv:2604.14853*, 2026.

Bowen Zuo and Yinglun Zhu. Strategic scaling of test-time compute: A bandit learning approach. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/d8d2c5f79c53a2a685dbdf93f78c5695-Abstract-Conference.html.

A PROOFS

Coverage without routability. Equal conditional means give $\sum_a \pi(a \mid X = x)\mu_a(x) = p(x)$ for every policy measurable with respect to X. Under conditional independence, all actions fail with probability $(1 - p(x))^A$; the complement is RCO. For $A > 1$ and $p \in (0, 1)$, $(1 - p)^A < 1 - p$.

Table 5: Marginal action accuracy (%) on every evaluation cell. Stochastic action means use all 16 outcomes per state; stop is the expected accuracy of uniform tie resolution in the two-response prefix. The fixed action was chosen only on disjoint fitting problems.

Model	Benchmark	Stop	Retry	Diversify	Critique	Decompose	Fixed choice
Qwen3-8B	MATH	96.00	95.45	95.85	94.80	95.80	Diversify
R1-Distill-7B	MATH	94.40	93.95	93.70	94.15	93.05	Critique
Phi-3.5-mini	MATH	43.20	44.85	44.05	46.35	43.40	Retry
Qwen3-8B	GPQA	45.83	46.31	49.43	34.61	53.12	Decompose
R1-Distill-7B	GPQA	26.14	25.90	27.37	24.53	27.32	Retry
Phi-3.5-mini	GPQA	30.30	26.99	30.45	31.91	29.07	Stop

Coupling bounds. For general utilities, $\max_a Y_a \geq Y_b$ for each b and $\max_a Y_a \leq \min\{1, \sum_a Y_a\}$; expectations give the two bounds. For Bernoulli marginals, nested success events attain the lower bound. Disjoint events attain the summed upper bound when $\sum_a \mu_a \leq 1$; arranging events to cover the unit interval attains one otherwise. Independent Bernoulli events give the product-coupling expression.

Plug-in optimism. Jensen's inequality applied to the convex maximum gives $\mathbb{E}\max_a \widehat{\mu}_a \geq \max_a \mathbb{E}\widehat{\mu}_a$. Also,

$$\max_a \widehat{\mu}_a - \max_a \mu_a \leq \max_a(\widehat{\mu}_a - \mu_a).$$

Each bounded m-sample mean is sub-Gaussian with variance proxy $1/(4m)$. For $\lambda > 0$, log-sum-exp therefore bounds the expectation on the right by $\log(A)/\lambda + \lambda/(8m)$. Optimizing λ gives $\sqrt{\log(A)/(2m)}$.

Split-selected value. Conditional independence implies $\mathbb{E}[\bar{Y}_a^E \mid S, X] = \mu_a(X)$. Conditional on S and X, the selection weights are fixed. Linearity then gives $\mathbb{E}[\sum_a w_a(S)\bar{Y}_a^E \mid S, X] = \sum_a w_a(S)\mu_a(X)$.

No finite-sample unbiased EAO estimator. For a fixed finite binary sample, the probability of every outcome configuration is a multivariate polynomial in the action means. The expectation of any finite-valued statistic is a finite linear combination of these probabilities and is therefore a polynomial. If such an expectation equaled EAO everywhere, restricting all but two means to zero would produce a polynomial equal to $\max(\mu_1, \mu_2)$ on $[0, 1]^2$. On either open half-square this maximum equals a different linear polynomial. A polynomial equal to one of them on an open set equals it everywhere, a contradiction.

B Action definitions and failure handling

The two prefix responses are immutable. Retry uses the unchanged problem prompt. Diversification receives only unverified answer-mode counts and asks for a substantially different derivation. Critique receives those counts and a deterministic representative trace marked as untrusted. Decomposition receives no answer history and requests explicit subgoals and independent verification. No action prompt contains a correctness marker or reference answer. A deterministic context fit shortens only the optional critique trace when necessary. Missing or unresolved final answers are incorrect, and no row is removed on the basis of its outcome.

A concrete audited state. One MATH-500 problem asks for $x + y + z$ given positive variables satisfying $xy = 24$, $xz = 48$, and $yz = 72$; its reference answer is 22. For Phi-3.5-mini, the two-response prefix was incorrect, so stop had utility 0. Across 16 fresh outcomes, critique, decomposition, diversification, and retry succeeded 4, 6, 7, and 7 times, respectively. Thus no single action's empirical success rate exceeded $7/16$, while at least one action succeeded at 15 of the 16 matched replicate indices. This state illustrates why realized coverage and the value of choosing one action before observing its outcome are different objects; the aggregate conclusions use all states rather than this example.

C GENERATION AND SCORING DETAILS

Pinned sources. The checkpoints and their revisions are:

- `Qwen/Qwen3-8B`: `b968826d9c46dd6066d109eabc6255188de91218`.
- `deepseek-ai/DeepSeek-R1-Distill-Qwen-7B`: `916b56a44061fd5cd7d6a8fb632557ed4f724f60`.
- `microsoft/Phi-3.5-mini-instruct`: `2fe192450127e6a83f7441aef6e3ca586c338b77`.

Qwen3 uses thinking mode; all three use bfloat16 and tensor-parallel size one. The other pinned sources are:

- MATH-500 test split, `HuggingFaceH4/MATH-500`: `6e4ed1a2a79af7d8630a6b768ec859cb5af4d3be`.
- GPQA-Diamond, `Idavidrein/gpqa`: `633f5ee89ab8ad4522a9f850766b73f62147ffdd`.
- Semantic encoder, `Qwen/Qwen3-Embedding-0.6B`: `97b0c614be4d77ee51c0cef4e5f07c00f9eb65b3`.

Answer options are permuted once per problem with split seed `20270907`.

Sampling. Generation uses temperature 0.6, top-p 0.95, a 32,768-token context limit, and at most 24,576 new tokens. Each state has two initial responses and 16 fresh outcomes for each of retry, diversification, critique, and decomposition. The base seed is `20270000`; action seeds start from `40420000`, with offsets 100,000 for diversification, 200,000 for critique, and 300,000 for decomposition. The per-response seed adds $1000 \cdot$ source index $+$ sample index to its base. Optional critique traces are capped at 6,000 tokens before deterministic context fitting.

Prompts. The MATH instruction, followed by the problem, is:

> Solve the following problem carefully. Show your reasoning, and finish with exactly one final answer in `\boxed{}`.

The GPQA instruction instead requires exactly one boxed answer-choice letter from A through D; a constrained one-token readout follows the complete response. The action-specific instructions are:

> **Diversification:** Solve the problem using a substantially different approach from those that could have produced the observed answer modes. The observed answers are unverified and may all be wrong. Do not vote among them or assume that frequency indicates correctness. Seek an independent derivation, check it carefully, and finish with the final answer in `\boxed{}`.
>
> **Critique:** Treat the supplied attempt as untrusted. Identify its first substantive error or unjustified step, then solve the original problem independently with that issue corrected. Do not preserve its final answer merely for consistency. Check the new derivation and finish with the final answer in `\boxed{}`.
>
> **Decomposition:** Before calculating, decompose the problem into explicit subgoals and identify the constraints that the final answer must satisfy. Solve the subgoals, recombine them, and perform an independent verification. Finish with the final answer in `\boxed{}`.

Retry uses the unchanged task instruction. Diversification receives only unverified answer-mode counts; critique also receives a deterministic representative trace marked untrusted. Decomposition receives no answer history.

Scoring. MATH uses equivalence-based answer scoring; GPQA uses the permuted option choice. Missing, unresolved, exceptional, and truncated answers remain incorrect in the full denominator. Stop is the expected accuracy under uniform tie resolution over the two-response prefix.

Table 6: Cross-action alignment sensitivity (percentage points). Product coupling averages RCO over independent action permutations and uses independent selection panels per action. Delta is the matched-index gap minus the product-coupled gap.

Model	Benchmark	Product gap (95% CI)	Matched-index gap	Delta
Qwen3-8B	MATH	1.19 [0.45, 2.14]	1.23	+0.04
R1-Distill-7B	MATH	1.51 [0.74, 2.54]	1.59	+0.08
Phi-3.5-mini	MATH	10.53 [8.37, 12.79]	10.62	+0.09
Qwen3-8B	GPQA	8.28 [6.33, 10.41]	8.56	+0.28
R1-Distill-7B	GPQA	8.17 [5.86, 10.67]	8.55	+0.38
Phi-3.5-mini	GPQA	18.74 [16.14, 21.33]	18.51	-0.23

Table 7: Cell-wise matched-panel RCO-minus-V_m gaps (percentage points) under declared analysis variations. Parsed-only results change the denominator; the other columns retain all evaluation problems. Token-normalized values are percentage points per 1,000 mean completion tokens.

Model	Benchmark	Deterministic ties	No stop	Parsed only	Per 1k tokens
Qwen3-8B	MATH	0.55	1.01	0.16	0.02
R1-Distill-7B	MATH	1.00	1.95	0.58	0.03
Phi-3.5-mini	MATH	8.28	10.94	11.25	7.37
Qwen3-8B	GPQA	6.45	9.83	8.56	0.35
R1-Distill-7B	GPQA	6.74	11.81	8.55	0.56
Phi-3.5-mini	GPQA	14.63	20.63	18.51	10.98

D Additional statistical details

For the common-panel sensitivity analysis, the split-panel estimator enumerates every subset of eight indices from 16; its complement is the evaluation panel. When several actions share the maximum selection mean, their evaluation means receive equal weight. Stability uses the inner product of the two tied-maximizer weight vectors. The 8/8 stability value is exact over complementary panels; smaller panels use deterministic Monte Carlo over disjoint panels. Posterior analyses use 10,000 draws. Primary intervals use 10,000 problem-bootstrap replicates, and action comparisons use 50,000 sign-randomization draws followed by Holm correction within each cell.

The cross-action alignment analysis averages RCO exactly under independent permutations conditional on empirical action distributions. For split-selected value, five independent runs each use 100,000 joint partitions, with a separate uniform 8/8 split for every stochastic action. The largest deviation of a cell estimate from its five-seed average is below 0.1 percentage points.

Controlled coupling simulation. We stress-test the coupling result in a second simulation that crosses two to eight actions, one to 16 outcomes per panel, 100 or 500 problems, three base success rates, two action-separation levels, and coupling strengths 0, 0.5, and 0.9. Each outcome is drawn either independently across actions or from nested Bernoulli success events, with the coupling strength giving the mixture probability. Across all 540 conditions and 2,000 datasets per condition, RCO is never below EAO and decreases monotonically as positive coupling increases. The largest absolute Monte Carlo bias of split-panel evaluation for its finite selection target is 0.21 percentage points. For four actions, eight outcomes per panel, 500 problems, base success 0.60, and a 0.15 best-action advantage, RCO minus EAO contracts from 23.40 points under independence to 2.34 points at coupling 0.9. This simulation verifies the dependence mechanism; the empirical coupling analysis uses the observed action marginals and is reported separately.

E Held-out routing details

All routers are fit on the problem splits reported in Section 7 and evaluated once on disjoint problems. Ridge penalties in $\{0.1, 1, 10, 100\}$ and semantic nearest-neighbor counts in $\{3, 5, 10, 20\}$ are selected by five-fold cross-validation on fitting problems, maximizing chosen-action value on the held-out fitting folds. Prompt-action targets are Jeffreys-smoothed empirical success rates for

Table 8: Cell-wise uncertainty of empirical best-action labels. Agreement is tie-aware across disjoint selection panels; "unresolved" is the fraction of states whose largest Jeffreys-posterior action-optimality probability is below 0.8.

Model	Benchmark	1/1	2/2	4/4	8/8	Unresolved
Qwen3-8B	MATH	20.7	21.1	21.6	22.2	3.2
R1-Distill-7B	MATH	20.8	21.4	22.3	23.6	4.8
Phi-3.5-mini	MATH	23.9	25.9	30.2	34.9	48.0
Qwen3-8B	GPQA	28.4	33.4	41.2	48.4	33.3
R1-Distill-7B	GPQA	22.7	25.3	30.1	36.9	66.7
Phi-3.5-mini	GPQA	26.4	31.8	40.6	49.8	54.5

Table 9: Held-out policy differences from the fit-selected fixed action, pooled by benchmark (percentage points). Intervals use paired problem bootstraps. Each router was tuned only on fitting problems; no router was selected using these evaluation outcomes.

Action bank	Benchmark	Router	Difference	95% CI
Prompt	MATH	Structured ridge	+0.02	[-1.03, +1.03]
Prompt	MATH	Signed-hash ridge	+0.40	[-0.98, +1.87]
Prompt	MATH	Semantic ridge	-0.72	[-2.13, +0.80]
Prompt	MATH	Semantic kNN	-0.75	[-2.15, +0.80]
Prompt	GPQA	Structured ridge	-0.25	[-2.07, +1.61]
Prompt	GPQA	Signed-hash ridge	-1.06	[-3.35, +1.25]
Prompt	GPQA	Semantic ridge	-3.12	[-5.38, -0.96]
Prompt	GPQA	Semantic kNN	-1.29	[-3.22, +0.58]
Model	MATH	Structured ridge	+0.00	[+0.00, +0.00]
Model	MATH	Signed-hash ridge	-0.30	[-0.80, +0.00]
Model	MATH	Semantic ridge	-1.20	[-2.70, -0.15]
Model	MATH	Semantic kNN	+0.00	[+0.00, +0.00]
Model	GPQA	Structured ridge	-9.19	[-14.58, -3.98]
Model	GPQA	Signed-hash ridge	-7.34	[-12.64, -2.32]
Model	GPQA	Semantic ridge	-6.63	[-11.41, -2.27]
Model	GPQA	Semantic kNN	-5.30	[-10.80, -0.05]

stochastic actions and the tie-aware fit value for stop. Structured prompt features contain benchmark metadata, problem-length descriptors, prefix lengths, parsed-answer agreement, answer-mode counts, and simple lexical indicators. The signed-hash variant adds 256 fixed text-hash coordinates. Neither representation uses a reference answer, correctness marker, or evaluation outcome.

Semantic routers use 1,024-dimensional problem embeddings from Qwen3-Embedding-0.6B at a pinned checkpoint revision. The semantic ridge head concatenates these embeddings with the structured features, while semantic nearest neighbors uses cosine similarity. GPQA embeddings are treated as restricted derived data and are not included in the PDF. A leakage test verifies that changing reference answers or correctness labels cannot change any router feature.

For model routing, one fresh retry distribution from each model defines three actions. The fitting target uses the first eight outcomes per model on the 50 MATH or 66 GPQA problems shared by all models. Evaluation uses all 16 outcomes on 125 MATH or 132 GPQA problems. Thus model selection never exploits the larger model-specific MATH fitting sets used to choose prompt actions.