SAME UNCERTAINTY, DIFFERENT REMEDIES: FAILURE-AWARE ROUTING OF TEST-TIME COMPUTE

Anonymous authors
Paper under double-blind review

ABSTRACT

Adaptive test-time scaling decides where and how much inference compute to spend, but after an initial sampling batch the system must decide what the next computation should do. Scalar uncertainty can signal risk, yet it does not specify whether to retain the majority, verify competing candidates, or abstain. We study whether answer-distribution structure provides a routing state using answer counts alone. Across 1,577 GPQA question–model pairs, the distributions admit operational regimes with different candidate concentration and intervention profiles. This motivates a failure-aware router that keeps majority voting outside a contestation gate, verifies concentrated top-two contests with a sampler-specific screened verifier, and abstains on scattered cases. In the primary Qwen3-32B/GPQA setting, verification gains 35.3 percentage points on 17 two-mode flags. With 51 verifier calls, full-coverage accuracy rises from 54.5% to 58.7%; abstaining on 11 scattered flags yields 61.4% accuracy at 92.3% coverage. The router has the largest observed equal-budget point estimate, although its advantage over runner-up-share allocation is not statistically established. Verification utility also varies across sampler–verifier pairs. These results suggest that uncertainty identifies risk, structure identifies candidate availability, and realizing that opportunity additionally requires a competent resolver. Routing should therefore follow the observed failure configuration and screen the resolver it depends on.

1 INTRODUCTION

Test-time scaling improves large language model reasoning by spending additional inference compute on sampling, search, refinement, or verification. A growing line of work makes this process adaptive rather than using a uniform budget. These methods allocate compute according to estimated difficulty, confidence, agreement, or intermediate reasoning signals (Snell et al., 2024; Komiyama et al., 2026; Huang et al., 2026; Li et al., 2026; Hariri et al., 2026). Recent methods further coordinate generation, ranking, verification, and stopping under a budget (Dughmi et al., 2026; Vilas et al., 2026; Zhou et al., 2026). These advances sharpen the questions of how much compute to spend and where to spend it. After an initial batch of reasoning trajectories has produced multiple candidate answers, however, the system faces another decision. What should the next unit of compute do? It may draw more samples, reflect on its current solution, verify competing candidates, generate a fresh answer, or stop and abstain. Uncertainty alone does not establish which action will have positive net value. An intervention can overturn a correct answer as well as repair an incorrect one. Adaptive test-time scaling therefore needs both risk signals and state information that is relevant to action choice.

Majority voting (MV) and self-consistency provide a natural setting for studying this problem. The initial sampling batch already exposes an empirical answer distribution (Wang et al., 2023; Chen et al., 2023). Prior work improves aggregation through pruning, reweighting, confidence signals, candidate selection, or refinement (Hong et al., 2025; Taubenfeld et al., 2025; Huang et al., 2024b; Petullo et al., 2026; Di et al., 2026; Wang et al., 2026a). Other work shows that consensus can reinforce systematic errors (Pan et al., 2026; Yan et al., 2026). For routing, the important distinction is not only how uncertain the vote is, but also how its mass is organized. A concentrated contest between two dominant answers can leave selection headroom if the correct answer is already among the leading candidates. A more dispersed distribution may offer weaker support for top-two correction. Yet candidate availability alone is insufficient. A fallible verifier may repeat the sampler's

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.

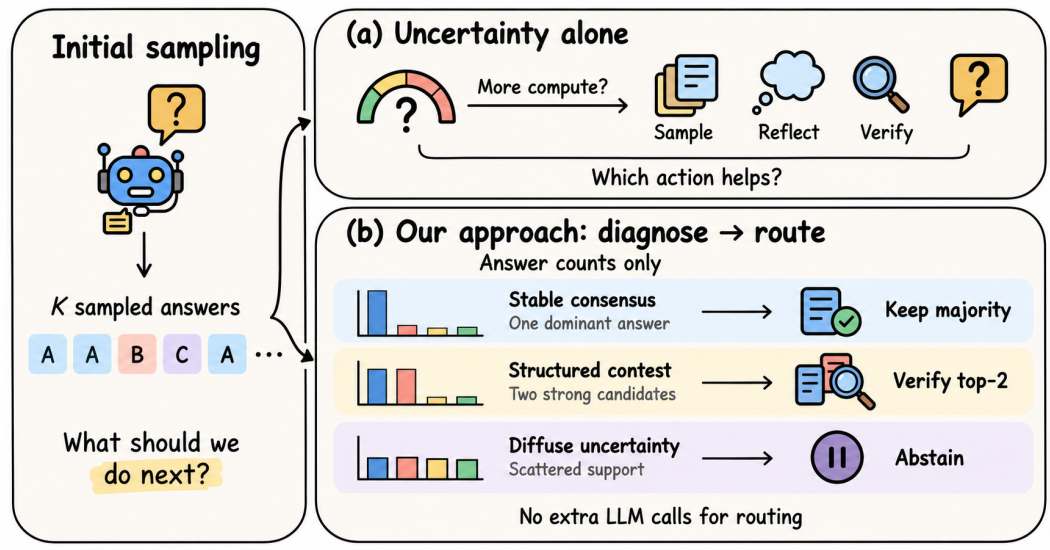

Figure 1: **From uncertainty to next-action routing.** An initial sampling batch supplies both a risk signal and an answer-count configuration. The latter suggests whether to retain the current answer, resolve a concentrated contest, or abstain on scattered support. The three rows illustrate intervention opportunities; they are not exact assignments of the deployed rule. Verification also requires a suitable, offline-screened resolver. Count-based diagnosis adds no model calls, while verification and calibration have separate costs. Section 6 specifies the policy and Section 7 its limits.

preference or overturn a correct majority, so useful verification also depends on the sampler–verifier pair. This motivates two separable questions. Is a correct candidate available, and can the chosen resolver identify it with positive net value? We therefore ask whether the first-round answer distribution can be a practical state representation for routing the next test-time action.

We study this question using answer counts alone. For each question, we summarize the initial samples by majority share, runner-up share, top-two mass, and normalized answer entropy. A descriptive Dirichlet–multinomial mixture provides an interpretable view of the answer-count space with three operational regimes. We refer to these regimes as stable consensus, structured contest, and diffuse uncertainty. They summarize a continuous distributional space rather than statistically identified or universally discrete failure types. Indeed, the mixture fit does not uniquely select three components. We then evaluate candidate availability through top-two selection headroom. We evaluate interventions by their paired change relative to MV, explicitly counting both rescued errors and damaged correct answers. The tested actions include additional sampling, reflection, same-source verification, screened verification, fresh generation, and abstention. The mixture model is used for diagnosis rather than deployment. The deployed rule uses only observed counts, while verifier choice is calibrated offline for a particular sampler–verifier pair. This design asks a narrower question than whether one uncertainty score is universally superior. Given an observed answer configuration, which additional action has useful value in the evaluated setting?

Our claim is deliberately narrower than proposing a universal failure taxonomy or a universally optimal router. The three regimes organize the observed answer-count space, while the deployed thresholds are simple count-based proxies rather than uniquely identified decision boundaries. Likewise, verification is treated as pair-specific because a different model is not automatically a useful resolver. We therefore evaluate routing together with its failure boundaries, including settings where top-two selection has little headroom or verifier utility does not transfer. The goal is to test whether inexpensive distributional state can support action-aware compute allocation, not to establish that one structural feature universally dominates scalar uncertainty.

Our contributions are threefold.

1. **Failure-aware formulation and count-based diagnosis.** We formulate post-sampling adaptive test-time scaling as a next-action routing problem. First-round answer counts provide a state signal without model calls beyond the initial sampling batch. Across 1,577 GPQA question–

model pairs from seven model families, a descriptive mixture yields three operational regimes. AIME exposes a boundary where top-two routing misses dispersed failures.

2. **Recoverability through candidate availability and resolver competence.** We separate whether the correct answer is already available among the leading samples from whether a verifier can select it without excessive damage. In structured contest, top-two coverage is 76.9% compared with 48.6% MV accuracy, exposing selection headroom. The verification matrix further shows that realized gains depend on the sampler–verifier pair.
3. **A practical failure-aware router with explicit statistical boundaries.** We instantiate these observations as a count-only keep/verify/abstain policy with offline verifier screening. In the primary Qwen3-32B/GPQA pool, the policy raises observed full-coverage accuracy from 54.5% to 58.7%, a gain of 4.2 percentage points, using 51 verifier calls. It attains the largest observed equal-budget point estimate among the tested allocators. Its advantage over a strong runner-up-share allocator, however, is not statistically established.

2 Related Work

Adaptive test-time computation. Model cascades and adaptive inference policies allocate resources according to estimated difficulty, confidence, or intermediate signals (Chen et al., 2024; Snell et al., 2024; Komiyama et al., 2026; Huang et al., 2026; Li et al., 2026; Hariri et al., 2026). More recent approaches jointly organize generation, ranking, verification, and stopping (Dughmi et al., 2026; Vilas et al., 2026; Zhou et al., 2026). Our setting is the decision after an initial sampling batch. We examine answer counts as an inexpensive routing state and measure whether the selected action helps when its resolver is fallible. This complements adaptive budget allocation and action coordination with an empirical account of candidate availability and pair-specific verification value.

Aggregation and unreliable consensus. Self-consistency aggregates sampled solutions (Wang et al., 2023; Chen et al., 2023). Extensions prune trajectories, use confidence or trajectory information, and select or refine candidate answers (Fu et al., 2026; Hong et al., 2025; Taubenfeld et al., 2025; Zhang et al., 2026; Huang et al., 2024b; Petullo et al., 2026; Di et al., 2026; Wang et al., 2026a). Consensus can nevertheless reinforce systematic errors, motivating selective pseudo-labels and alternative training signals (Pan et al., 2026; Yan et al., 2026; Wang et al., 2026b). Debate studies also examine voting, consensus-free decisions, and when to overturn a majority (Choi et al., 2025; Cui et al., 2026; He et al., 2026). We use MV as the starting prediction for evaluating subsequent actions, rather than as the central object to improve.

Uncertainty and selective prediction. Semantic uncertainty and cross-model disagreement enrich uncertainty estimation (Kuhn et al., 2023; Hamidieh et al., 2026); Verify When Uncertain adapts consistency checking for hallucination detection (Xue et al., 2026). Selective prediction trades answer coverage for accuracy (El-Yaniv & Wiener, 2010). Our question is whether the configuration of candidate support helps choose an action after risk has been detected. Runner-up share is both a scalar and a count-derived structural feature, so we compare against strong scalar rules and explicitly test the incremental value of top-two concentration.

Correction and verification. Reflection, self-refinement, debate, model judges, and process supervision provide different ways to revise or assess an answer (Shinn et al., 2023; Madaan et al., 2023; Du et al., 2024; Zheng et al., 2023; Lightman et al., 2024; Dhuliawala et al., 2024). Their value depends on the feedback and evaluator (Huang et al., 2024a; Wu et al., 2024; Sharma et al., 2024). We therefore count damaged correct answers as well as rescued errors. A correct candidate and a different verifier are each insufficient; useful verification requires a favorable net outcome for the particular sampler–verifier pair.

Concurrent work. Disagreement-guided strategy routing (Lin et al., 2026) also formulates test-time scaling as an instance-level routing problem, using output disagreement to select among lightweight resolution, voting, and rewriting. Our focus differs in reading the answer-count configuration as a state that separates candidate availability from resolver competence, and in measuring paired rescue and damage for verification and abstention rather than selecting a scaling strategy. ARBITER (Cai et al., 2026) analyzes majority-vote failures through reasoning basins and reports the same kind of recoverable top-two headroom, recovering part of it from the base model's hidden states. We ask what that headroom is worth under a count-only state and a deployable black-box verifier, and

find that it does not convert into realized accuracy: the resolver, not the structure, is the binding constraint.

3 ROUTING FROM THE FIRST SAMPLING BATCH

3.1 ANSWER COUNTS AS AN OBSERVABLE STATE

For each question, the sampler generates K reasoning trajectories at temperature 0.7. Let n be the number of valid extracted answers, with sorted counts $\mathbf{n} = (n_1, n_2, \ldots)$ and corresponding answers $a_1, a_2, \ldots$ MV returns a_1; throughout, MV includes plurality selection. The observed state is summarized by

$$p_1 = \frac{n_1}{n}, \qquad \eta = \frac{n_2}{n}, \qquad t_2 = \frac{n_1 + n_2}{n}, \qquad H = -\frac{\sum_{i=1}^{d} p_i \log p_i}{\log d}, \tag{1}$$

where d is the number of distinct extracted answers, $p_i = n_i/n$, and $H{=}0$ for $d{=}1$. We take $n_2{=}0$ when only one answer is observed. All shares use n, not the nominal budget K. Computing these features requires neither token probabilities nor additional model calls.

The state records how candidate support is distributed; it does not reveal correctness. We use *failure-aware* to mean that routing is informed by observable configurations associated with different intervention opportunities. Gold answers are used only in offline analysis and calibration.

3.2 CANDIDATE AVAILABILITY AND RESOLVER COMPETENCE

Let $R_k(q)$ indicate whether the gold answer to question q appears among the top k sampled answers. The top-two selection headroom is

$$h(q) = R_2(q) - R_1(q). \tag{2}$$

Its mean is the maximum accuracy gain obtainable by perfect selection between the leading two candidates. It measures availability, not the performance of a real verifier. For an answer-producing action a, we instead evaluate the paired change

$$g(a \mid S) = \frac{1}{|S|} \sum_{q \in S} [\mathbf{1}\{a(q) \text{ correct}\} - \mathbf{1}\{a_1(q) \text{ correct}\}] = \frac{N_{\text{rescue}} - N_{\text{damage}}}{|S|}. \tag{3}$$

The action set includes keeping MV, further sampling, reflection, same-source verification, screened verification, fresh generation, and abstention. Different actions are evaluated on different available subsets S; their denominators are always stated. Abstention is evaluated through accuracy and coverage rather than being counted as a corrected answer. These paired measurements characterize the tested protocols; they do not identify causal effects of the descriptive regime assignments.

3.3 EVALUATION SETTING

We evaluate nine checkpoints from seven model families: Qwen3-32B, Qwen3-8B, Qwen3-30B-A3B, Llama-4-Scout-17B, GLM-4.7-Flash, Gemma3-27B, Phi-4, InternLM3-8B, and DeepSeek-R1-Distill-Qwen-32B (Yang et al., 2025; AI at Meta, 2025; Z.ai, 2026; Gemma Team, 2025; Abdin et al., 2024; Shanghai AI Laboratory, 2025; DeepSeek-AI et al., 2025). The primary setting is Qwen3-32B on the 198-question, four-option GPQA-Diamond benchmark with $K{=}32$ (Rein et al., 2024). The descriptive analysis pools 1,577 GPQA question–model pairs. MMLU-Pro, ScienceQA, MATH, GSM8K, and 45 AIME problems provide transfer and boundary analyses (Wang et al., 2024; Lu et al., 2022; Hendrycks et al., 2021; Cobbe et al., 2021).

The primary routing pool contains 143 questions with at least 20 valid letter outputs. A separate question-level split assigns 98 GPQA questions to discovery and 100 to test; model fitting and verifier screening use discovery data in the frozen evaluation. Full-pool analyses, frozen test results, and older intervention subsets are reported separately. Appendix B specifies their construction, and Appendix F describes uncertainty estimates and tests.

Region	Pairs	$\mathbb{E}[p_1]$	$\mathbb{E}[t_2]$	MV	Top two	Runner-up	$\mathbb{E}[d]$
Stable consensus	546	0.98	1.00	75.3%	78.4%	3.1%	1.3
Structured contest	592	0.76	0.96	48.6%	76.9%	28.2%	2.6
Diffuse uncertainty	439	0.51	0.80	34.9%	61.5%	26.7%	3.7
AIME: overall	45	—	—	40.0%	40.0%	0.0%	11.4
Contest-like	25	—	—	36.0%	36.0%	0.0%	14.6
Diffuse-like	15	—	—	26.7%	26.7%	0.0%	9.6

Table 1: **Candidate availability across descriptive regimes.** GPQA uses hard assignments from the three-component fit to 1,577 question–model pairs. $\mathbb{E}[p_1]$ and $\mathbb{E}[t_2]$ are component posterior means, so they are undefined for the AIME rows, which are assigned under the frozen GPQA fit rather than refitted. MV and runner-up columns report accuracy; "Top two" reports gold-answer coverage. Runner-up accuracy equals top-two selection headroom before rounding. On AIME, the frozen fit assigns 5/25/15 questions to stable/contest-like/diffuse-like regions. No question has a correct runner-up, so the runner-up column is 0.0%.

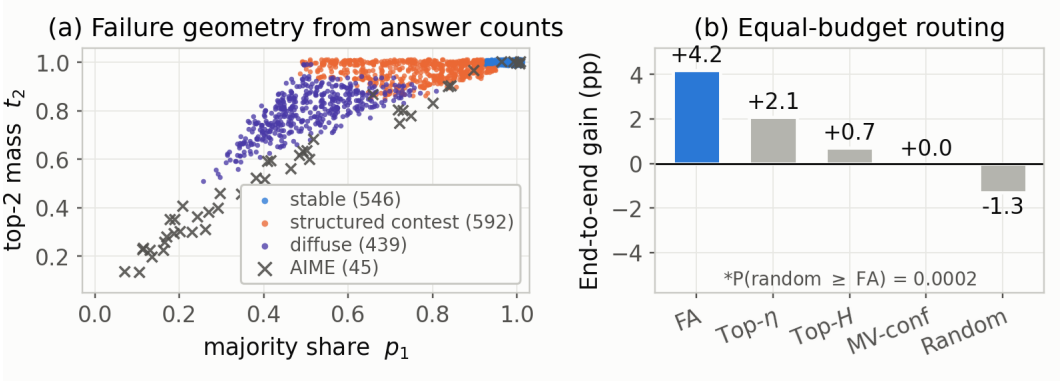

Figure 2: **Distributional state and observed routing value.** (a) The 1,577 GPQA question–model pairs form a continuous answer-count space; colors show the descriptive three-component assignments. AIME illustrates a change in the answer space. (b) On the primary pool, allocating 51 verifier calls by the count gate has the largest full-coverage gain, $+4.2$ pp. This is a point-estimate ordering; direct paired comparisons with runner-up-share and entropy allocation are inconclusive. The random-allocation reference addresses a different null.

4 What Does the Answer Distribution Reveal?

Three operational regimes. We fit a Dirichlet–multinomial mixture to the sorted counts,

$$z \sim \mathrm{Cat}(\boldsymbol{\pi}), \qquad \mathbf{n} \mid z \sim \mathrm{DirMult}(\boldsymbol{\alpha}_z), \qquad \hat{z}(q) = \arg\max_z P(z \mid \mathbf{n}). \tag{4}$$

Sorting makes this a descriptive fit to ranked counts rather than a generative model of fixed answer identities. BIC decreases from 14,348 at two components to 13,881 at six; it does not identify three components as optimal. We use three to summarize stable consensus, structured contest, and diffuse uncertainty (Figure 2). The mixture organizes the analysis and is not run by the deployed router.

Different candidate configurations. In stable consensus, MV accuracy is 75.3% and the runner-up adds only 3.1 pp of selection headroom. In structured contest, MV accuracy falls to 48.6%, but top-two coverage remains 76.9%. In diffuse uncertainty, these quantities fall to 34.9% and 61.5% (Table 1). Diffuse uncertainty still has substantial average headroom: 26.7 pp, compared with 28.2 pp in structured contest using unrounded values. Thus, lower concentration does not imply that verification is impossible. It motivates testing whether a particular resolver can use the available candidates reliably; Appendix F.5 shows at the flag level that the availability contrast survives conditioning on η.

Descriptive regions and action gates differ. The structured-contest component includes many questions with modest disagreement. Among 361 pooled pairs satisfying $\eta \geq 0.3$, 147 are assigned to structured contest and 214 to diffuse uncertainty. A separate four-component model supplies the contestation score $C(q) = P(\text{contest} \mid \mathbf{n})$, whose association with η is strong on the primary

Region	MV	Resampling	Reflection	Same-source verification	Screened verification	Off-gate verification
Stable (n=53)	79.2%	+0.0	−9.1 (44)	—	—	−33.3 (3)
Contest (n=94)	52.1%	≈ 0	−9.0 (67)	−15.4 (13)	+30.8 (13)	−20.8 (24)
Diffuse, $t_2 \geq 0.9$ (n=7)	28.6%	≈ 0	+0.0 (4)	+50.0 (2)	+50.0 (2)	−40.0 (5)
Diffuse, $t_2 < 0.9$ (n=44)	36.4%	≈ 0	−3.7 (27)	+33.3 (9)	+12.5 (8)	−29.2 (24)

Table 2: **Intervention value depends on both the question subset and resolver.** Stable and Contest are descriptive mixture regions; the two Diffuse rows are that one region split by the operational t_2 gate, so the mixture regions and the gate subsets are not interchangeable. Primary-sampler action changes are in pp relative to MV on the same evaluated questions, with sample sizes in parentheses. The MV column covers the entire named regime and is not necessarily an intervention's paired baseline. Screened and off-gate verification use Gemma3-27B with three calls, on flagged and unflagged questions respectively. Resampling compares K=16 and K=32 over five subsampled halves. Dashes indicate unreported evaluations.

sampler (ρ=0.98). Its 0.5 crossing near η=0.30 motivates a simple count gate. Within this gate, t_2 distinguishes concentrated two-mode flags from scattered flags: the η=0.30 crossing is fitted from the mixture, whereas the t_2=0.9 cutoff is a simple operating point rather than an independently optimized boundary (Appendix F). These flags are not interchangeable with the mixture regimes.

More samples need not resolve selection. On 141 paired primary-sampler questions, MV accuracy remains around 54–55% from K=16 onward, while top-two coverage is approximately 73–76%. Additional samples may expose a correct candidate without making it the most frequent answer (Appendix B). This gap motivates an action that changes selection rather than only increasing the sample count.

5 Which Interventions Have Positive Value?

Table 2 evaluates the second requirement for useful intervention: resolver competence. Every change is paired on the evaluated intervention subset. The regional MV column provides context and need not equal that subset's baseline.

Retaining a stable answer can be preferable. On 44 stable-consensus questions, reflection rescues one error but damages five correct answers, losing 9.1 pp. Resampling has little observed effect. These results favor keeping MV for this evaluated region; they do not make consensus a correctness certificate.

A contest creates an opportunity, not a guarantee. On 13 flagged questions assigned to structured contest, screened verification rescues five errors and damages one correct answer, gaining 30.8 pp. Same-source verification loses 15.4 pp on its evaluated subset, while reflection loses 9.0 pp. The same screened verifier loses 20.8 pp when applied to unflagged questions in the descriptive contest region. Both question selection and resolver choice therefore matter.

Concentrated and scattered flags have different observed gains. In the primary routing pool, 28 questions satisfy $\eta \geq 0.3$. Verification gains 35.3 pp on the 17 with $t_2 \geq 0.9$ and 0 pp on the 11 with $t_2 < 0.9$. The former have seven rescues and one damage; the latter have two of each. This small-sample contrast motivates different actions, but the groups are not matched on η and their gain difference is not significant (p=0.12). Appendix F tests the incremental contribution of structure directly.

Resolver competence is pair-specific. In a separate pooled analysis of recoverable MV errors, same-source verification rescues 21.7%, while Gemma3-27B rescues 19/27 (70.4%) on the standard GPQA targets. Recoverability here conditions on the gold answer being one of the verifier's two candidates. The full matrix shows that changing model families is insufficient: as a verifier, Gemma3-27B helps the InternLM3-8B sampler but adds nothing for DeepSeek-R1-Distill, and on the Qwen3-30B-A3B sampler Phi-4 yields positive net value whereas Qwen3-32B yields negative value (Appendix A). We call a verifier *screened* if it has been evaluated on labeled calibration questions for its intended sampler.

Allocator	Rescue/ damage	Selected Δ	Correct/ 143	Pool Δ	Paired p
Failure-aware gate	7/1	+35.3	84/143	**+4.2**	0.070
Top 17 η	6/3	+17.6	81/143	+2.1	0.508
Top 17 H	4/3	+5.9	79/143	+0.7	1.000
Lowest 17 p_1	4/4	+0.0	78/143	+0.0	1.000
Action ablations and reference baselines					
Verify two-mode flags (17)	7/1	+35.3	84/143	+4.2	0.070
Verify scattered flags (11)	2/2	+0.0	78/143	+0.0	1.000
Select runner-up (17)	9/5	+23.5	82/143	+2.8	0.424
Random (50,000 draws)	—	—	mean 53.2%	−1.3	—
Uniform K=32→48 (n=118)	—	—	—	+0.8	—

Table 3: **Equal-budget verification allocation and action ablations.** The first four policies select 17 of the same 143 eligible questions, each using 51 Gemma3-27B calls. Unselected questions retain MV; the common baseline is 78/143 (54.5%). Changes are in pp. Paired p-values are two-sided exact McNemar tests against MV, not between allocators. Scattered-only verification uses 33 calls; direct runner-up selection uses none. Random allocation samples 17 questions uniformly. Uniform resampling is a separate comparison on 118 questions.

6 A Count-Based Router and Its Measured Value

6.1 Policy and calibration

With a suitable verifier available, the deployed policy is

$$\pi(\mathbf{n}) = \begin{cases} \text{keep MV}, & \eta < 0.3, \\ \text{verify the top two}, & \eta \geq 0.3 \text{ and } t_2 \geq 0.9, \\ \text{abstain}, & \eta \geq 0.3 \text{ and } t_2 < 0.9. \end{cases} \tag{5}$$

Verification uses M=3 calls at temperature 0.3. The verifier is Gemma3-27B, selected using discovery questions. The count rule requires no mixture inference or labels at deployment. Appendix F reports the retrospective sensitivity of the t_2 cutoff. Screening is an offline cost and the chosen verification calls are an online cost. If screening rejects the verifier, the policy keeps MV on two-mode flags and retains abstention on scattered flags.

Calibration evidence is limited. On the discovery split, Gemma3-27B rescues all three evaluated recoverable errors, versus one of four for the best tested same-source alternative. A separately reported frozen-model test subset of 13 questions gains 30.8 pp (bootstrap 95% CI $[7.7, 53.8]$). This subset differs from the eligible test pool used for matched-budget allocation below. Appendix F separates the verifier analyses.

6.2 Equal-budget allocation

The gate selects 17 of the 143 primary-pool questions for verification, using 51 calls, or 0.36 additional generations per question. For the full-coverage comparison, every unverified question retains its MV answer, including scattered flags. The comparator policies select 17 questions from the same pool, use the same verifier and protocol, and reuse the same verifier outputs.

Verification increases full-pool accuracy from 78/143 (54.5%) to 84/143 (58.7%). The 4.2 pp gain has a bootstrap 95% CI of $[0.7, 7.7]$, while the exact paired test against MV gives p=0.070; the two procedures summarize uncertainty differently and need not agree near the decision boundary (Appendix F). Runner-up-share allocation gains 2.1 pp, entropy 0.7 pp, MV-confidence allocation 0.0 pp, and random allocation averages −1.3 pp (Table 3). Direct paired comparisons do not establish superiority over runner-up-share or entropy allocation; the comparison with runner-up share gives p=0.375.

The frozen eligible test pool provides a further check. At 27 calls, both the count gate and top-η allocation answer 40/67 questions correctly, compared with 36/67 for MV. The gate has four rescues and no damage, whereas top-η has five rescues and one damage. The tie does not demonstrate incremen-

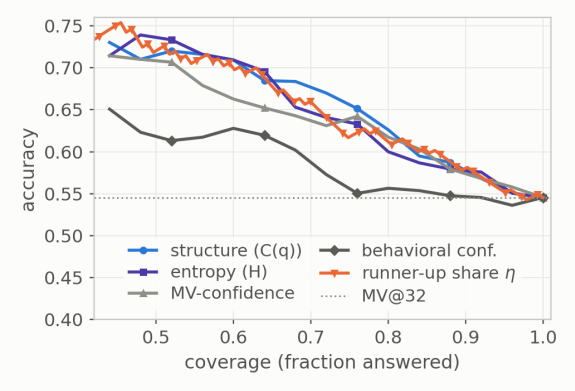

Figure 3: **Risk ranking without verification.** Accuracy–coverage curves on the 143-question primary pool retain MV on answered questions. $C(q)$, η, entropy, and majority share use counts; behavioral confidence uses trajectory token log-probabilities. The $C(q)$ and η curves stay within approximately 4 pp. Labels evaluate the curves and are not used to rank questions at inference time. The figure concerns abstention alone, not the verify-and-abstain policy.

tal accuracy value from t_2. Full-pool randomization, paired-policy tests, and frozen comparisons are distinguished in Appendix F.

How much does the verifier add? On the same 17 two-mode flags, MV answers five questions correctly. Selecting the runner-up directly answers nine, and screened verification answers eleven. Thus, the gate identifies useful alternative candidates, and verification adds two correct answers beyond the direct-switch rule: 1.4 pp over the full pool. This exploratory decomposition separates candidate selection opportunity from realized resolver value.

6.3 Abstention and Coverage

After verification, abstaining on the 11 scattered flags leaves 81/132 correct answers: 61.4% accuracy at 92.3% coverage. This selective result is distinct from the 58.7% full-coverage result. Abstaining on all 28 flags instead gives 60.9% accuracy at 80.4% coverage. Verification therefore retains answers on part of the flagged set while improving the observed selective outcome.

Count-derived abstention rankings also provide a useful reference (Figure 3). The $C(q)$ and η curves are similar and generally exceed entropy or majority-share ranking over the plotted range. These curves use MV on every answered question and invoke no verifier. They support count-based risk ranking without proving that a multidimensional state universally improves on a scalar score.

7 Transfer and Operating Limits

A new sampler requires a new resolver check. On InternLM3-8B, verifying two-mode flags adds 1.1 pp over the full pool. On Qwen3-8B, forced verification loses 0.5 pp; its selected policy therefore abstains on scattered flags without verification, reaching 50.6% accuracy at 84.4% coverage from a 49.0% full-coverage baseline. Appendix B separates verification from abstention for all three samplers. Pooled screened and unscreened results involve different pairs and questions, so their difference does not identify the benefit of screening itself.

Transfer of counts does not ensure transfer of action value. Without dataset-specific screening, top-two verification has no net gain on the MMLU-Pro routing pool and loses 0.4 pp on ScienceQA. Fresh generation and wider candidate lists likewise give mixed results (Appendix F.8). Exploration remains a candidate action, not an established remedy for scattered answers.

AIME exposes a missed-failure boundary. The frozen GPQA mixture assigns 40 of 45 AIME questions to contest-like or diffuse-like regions, yet none satisfies $\eta \geq 0.3$. The gold answer is never ranked second: it appears below rank two on 22.2% of questions and is never sampled on 37.8%. Top-two selection therefore has zero headroom. Small η can reflect errors dispersed over many numeric answers rather than stable consensus, so the present policy leaves these errors outside its

intervention gate. A single successful GSM8K case shows that concentrated disagreement can occur without option letters, but does not establish policy transfer. Detecting dispersed failures requires an additional rule; recovering them may require a different action.

8 Discussion and Conclusion

What the study establishes. The initial answer batch supports a practical separation between candidate availability and resolver competence. Candidate concentration identifies possible selection opportunities; paired intervention measurements determine whether a tested resolver realizes them. The resulting keep/verify/abstain policy illustrates how this information can guide the next computation. MV is the starting prediction and evaluation baseline for that routing problem.

What remains unresolved. The strongest equal-budget result comes from one sampler–dataset pair. The two-mode and scattered subsets are small and unmatched: their availability contrast survives conditioning on η, but matched-η differences in realized action value, and superiority over scalar allocation, are not established. The three regimes are descriptive, verifier calibration is small, and option-order effects remain unresolved. Generation-call budgets do not equalize tokens, latency, or FLOPs across models. AIME further shows that the current gate can miss errors in a large answer space. These limits motivate task-specific resolver calibration and common resource accounting.

Conclusion. Uncertainty signals that an answer may be unreliable. Answer-distribution structure provides a candidate state for deciding what to do next. Our evidence motivates routing test-time computation through both the available candidates and the competence of the chosen resolver, while locating the settings in which this simple count-based policy remains unproven or fails.

AI use statement

Generative AI tools assisted with research framing and hypothesis refinement, experimental design and planning, statistical analysis and interpretation, code for analyses and figures, literature retrieval, and manuscript and figure drafting and revision. They did not substitute for the model evaluations or statistical measurements reported here. All AI-assisted methodology, code, analyses, references, figures, and text were verified by the authors, who are responsible for the final content.

Reproducibility statement

The sampling, extraction, state representation, and policy are specified in Section 3 and Appendix B. The analysis and figure-generation scripts, the per-question statistics, and the data behind every figure will be released upon acceptance. Appendix F documents paired tests, bootstrap intervals, frozen comparisons, and retrospective sensitivity analyses. Reconstructing the allocation comparisons requires the per-question count vectors, gold labels, split assignments, verifier outputs, and output-validity flags; regenerating trajectories additionally requires the original prompts, checkpoints, and decoding configurations. These requirements distinguish reproducibility of the reported analyses from rerunning the underlying model generations.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

AI at Meta. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation. Meta AI Blog, `https://ai.meta.com/blog/llama-4-multimodal-intelligence/`, 2025.

Meng Cai, Lars Kulik, and Farhana Choudhury. ARBITER: Reasoning trajectory basins and majority vote failures in test-time sampling. *arXiv preprint arXiv:2605.26172*, 2026.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024. arXiv:2305.05176.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. Universal self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*, 2023.

Hyeong Kyu Choi, Xiaojin Zhu, and Sharon Li. Debate or vote: Which yields better decisions in multi-agent large language models? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Yu Cui, Hang Fu, Haibin Zhang, Licheng Wang, and Cong Zuo. Free-mad: Consensus-free multi-agent debate. In *Findings of ACL*, 2026.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 3563–3578, 2024. arXiv:2309.11495.

Qiwei Di, Kaixuan Ji, Xuheng Li, Heyang Zhao, and Quanquan Gu. Best-of-Majority: Minimax-optimal strategy for Pass@k inference scaling. In *International Conference on Learning Representations (ICLR)*, 2026.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of ICML*, 2024. arXiv:2305.14325.

Shaddin Dughmi, Mahdi Haghifam, and Yusuf Hakan Kalayci. Adaptive Generate-Rank-Verify: Inference-time search with costly verification. *arXiv preprint arXiv:2605.17609*, 2026.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010.

Yichao Fu, Xuewei Wang, Hao Zhang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. In *International Conference on Learning Representations (ICLR)*, 2026.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. Google DeepMind.

Kimia Hamidieh, Veronika Thost, Walter Gerych, Mikhail Yurochkin, and Marzyeh Ghassemi. Complementing self-consistency with cross-model disagreement for uncertainty quantification. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2604.17112.

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.

Chuan He, Zebin Chen, Zhengyi Yang, Shaobo Qiao, Mingchen Ju, Jiate Liu, Dong Wen, and Guanfeng Liu. Minority Sentinel: When to overturn majority voting in multi-agent LLM debates. *arXiv preprint arXiv:2606.29270*, 2026. Presented at AgentSearch Workshop @ SIGIR 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 *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Colin Hong, Xu Guo, Anand Chaanan Singh, Esha Choukse, and Dmitrii Ustiugov. Slim-SC: Thought pruning for efficient scaling with self-consistency. In *Proceedings of EMNLP*, pp. 34500–34517, 2025.

Chengsong Huang, Langlin Huang, Jixuan Leng, Jiacheng Liu, and Jiaxin Huang. CaTS: Calibrated test-time scaling for efficient LLM reasoning. In *International Conference on Learning Representations (ICLR)*, 2026.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations (ICLR)*, 2024a. arXiv:2310.01798.

Siyuan Huang, Zhiyuan Ma, Jintao Du, Changhua Meng, Weiqiang Wang, and Zhouhan Lin. Mirror-consistency: Harnessing inconsistency in majority voting. In *Findings of EMNLP*, pp. 2408–2420, 2024b. arXiv:2410.10857.

Junpei Komiyama, Daisuke Oba, and Masafumi Oyamada. Best-of-Infinity: Asymptotic performance of test-time LLM ensembling. In *International Conference on Learning Representations (ICLR)*, 2026.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations (ICLR)*,

2023. arXiv:2302.09664.

Pingzhi Li, Bairu Hou, Yun Zhu, Yihao Feng, Ke Ye, Tao Lei, Zhifeng Chen, Tianlong Chen, and Xianzhi Du. Adaptive Thinking: Large language models know when to think in latent space. In *International Conference on Learning Representations (ICLR)*, 2026.

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 (ICLR)*, 2024. arXiv:2305.20050.

Zhimin Lin, Yixin Ji, Jinpeng Li, Yu Luo, Dong Li, Junhua Fang, Juntao Li, and Min Zhang. When to vote, when to rewrite: Disagreement-guided strategy routing for test-time scaling. *arXiv preprint arXiv:2604.26644*, 2026.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2203.12511.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2303.17651.

Teng Pan, Yuchen Yan, Zixuan Wang, Ruiqing Zhang, Guiyang Hou, Wenqi Zhang, Weiming Lu, Jun Xiao, and Yongliang Shen. CoVerRL: Breaking the consensus trap in label-free reasoning via generator-verifier co-evolution. In *Proceedings of ACL*, 2026.

James Petullo, Sonny George, Dylan Cashman, and Nianwen Xue. VecCISC: Improving confidence-informed self-consistency with reasoning trace clustering and candidate answer selection. In *Findings of ACL*, 2026.

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 *Conference on Language Modeling (COLM)*, 2024. arXiv:2311.12022.

Shanghai AI Laboratory. InternLM3-8b-instruct model card. Hugging Face model card, `https://huggingface.co/internlm/internlm3-8b-instruct`, 2025.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2310.13548.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2303.11366.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in LLMs. In *Findings of ACL*, 2025.

Martina G. Vilas, Safoora Yousefi, Besmira Nushi, Eric Horvitz, and Vidhisha Balachandran. Tracing the traces: Latent temporal signals for efficient and accurate reasoning. In *International Conference on Learning Representations (ICLR)*, 2026.

Qibin Wang, Pu Zhao, Shaohan Huang, Fangkai Yang, Lu Wang, Furu Wei, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. Learning to refine: Self-refinement of parallel reasoning in LLMs. In *Findings of ACL*, 2026a.

Weiqin Wang, Yile Wang, Kehao Chen, and Hui Huang. Beyond majority voting: Towards fine-grained and more reliable reward signal for test-time reinforcement learning. In *Proceedings of ACL*, 2026b.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves Chain of Thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2203.11171.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024. arXiv:2406.01574.

Zhenyu Wu, Qingkai Zeng, Zhihan Zhang, Zhaoxuan Tan, Chao Shen, and Meng Jiang. Large language models can self-correct with key condition verification. In *Proceedings of EMNLP*, 2024. arXiv:2405.14092.

Yihao Xue, Kristjan Greenewald, Youssef Mroueh, and Baharan Mirzasoleiman. Verify When Uncertain: Beyond self-consistency in black box hallucination detection. *Transactions on Machine Learning Research*, 2026. arXiv:2502.15845.

Dong Yan, Jian Liang, Yanbo Wang, Shuo Lu, Ran He, and Tieniu Tan. What if consensus lies? selective-complementary reinforcement learning at test time. In *Proceedings of ACL*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Z.ai. GLM-4.7-Flash. Z.ai model card, `https://huggingface.co/zai-org/GLM-4.7-Flash`, 2026.

Kai Zhang, Jiayi Liao, Chengpeng Li, Ziyuan Xie, Sihang Li, and Xiang Wang. Chronos: Learning temporal dynamics of reasoning chains for test-time scaling. In *Findings of ACL*, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2306.05685.

Yan Zhou, Yue Ouyang, Kaiyang Zheng, and Suncheng Xiang. Coba: Cost-effective test-time scaling via compute-balanced routing. *arXiv preprint arXiv:2608.07424*, 2026.

A Verification Across Sampler–Verifier Pairs

Table 4 records every reported sampler–verifier combination. It separates a verifier's ability to correct errors from its net effect on all evaluated questions. The correction fractions use MV errors with valid verifier outputs under the corresponding analysis. They must not be interpreted as full-subset gains or automatically equated with rescue rates conditioned on recoverability.

The separate recoverable-error analysis conditions additionally on the gold answer being among the two candidates. Same-source verification rescues 10/46 errors (21.7%), while Gemma3-27B rescues 19/27 (70.4%) on the standard GPQA targets. The latter exceeds a 50% two-candidate reference in a one-sided exact binomial test (p=0.026). This reference concerns selection between available candidates; it is not a test of net accuracy improvement over MV. A verifier can rescue errors and still have low or negative net value by damaging correct answers.

Sampler	Verifier	Flags	Corr.	MV	Ver.	Δ
GPQA: same source						
Qwen3-32B	Qwen3-32B	24	2/9	.250	.250	+.000
Qwen3-8B	Qwen3-8B	41	2/11	.366	.293	−.073
Gemma3-27B	Gemma3-27B	30	3/10	.400	.267	−.133
Llama-4-Scout	Llama-4-Scout	16	0/3	.375	.313	−.062
GPQA: alternative verifiers						
Qwen3-32B	Gemma3-27B	24	7/8	.250	.417	+.167
Qwen3-8B	Gemma3-27B	41	7/11	.366	.293	−.073
Llama-4-Scout	Gemma3-27B	16	5/8	.375	.500	+.125
Qwen3-32B	Qwen3-8B	24	3/9	.250	.375	+.125
Llama-4-Scout	Qwen3-8B	16	4/8	.375	.500	+.125
Gemma3-27B	Qwen3-8B	30	3/10	.400	.433	+.033
Llama-4-Scout	Qwen3-32B	16	3/8	.375	.438	+.062
Qwen3-8B	Qwen3-32B	41	3/11	.366	.317	−.049
Gemma3-27B	Qwen3-32B	30	1/10	.400	.333	−.067
Qwen3-32B	GLM-4.7-Flash	24	3/5	.250	.333	+.083
Qwen3-8B	GLM-4.7-Flash	41	2/7	.366	.293	−.073
Gemma3-27B	GLM-4.7-Flash	30	2/9	.400	.433	+.033
Llama-4-Scout	GLM-4.7-Flash	16	3/7	.375	.438	+.062
Phi-4	Gemma3-27B	35	13/20	.314	.371	+.057
InternLM3-8B	Gemma3-27B	64	14/19	.281	.344	+.062
DeepSeek-R1-Distill*	Gemma3-27B	27	6/12	.370	.370	+.000
Qwen3-30B-A3B†	Gemma3-27B	16	3/8	.375	.250	−.125
DeepSeek-R1-Distill*	Qwen3-32B	27	1/12	.370	.296	−.074
Qwen3-30B-A3B†	Qwen3-32B	16	3/8	.375	.250	−.125
DeepSeek-R1-Distill*	Phi-4	27	2/12	.370	.259	−.111
Qwen3-30B-A3B†	Phi-4	16	7/8	.375	.688	+.312
ScienceQA: Qwen3-32B sampler						
Qwen3-32B	Qwen3-32B	35	2/10	.543	.486	−.057
Qwen3-32B	Gemma3-27B	35	6/10	.543	.457	−.086
Qwen3-32B	Qwen3-8B	35	4/10	.543	.514	−.029
MMLU-Pro: additional Qwen3-32B subset						
Qwen3-32B	Qwen3-32B	15	1/3	.733	.800	+.067
Qwen3-32B	Qwen3-8B	15	1/3	.733	.667	−.067

Table 4: **Pair-specific verification outcomes.** Accuracy and Δ are proportions on the stated flagged subsets. "Corrected" uses MV-error questions with valid outputs, whereas accuracy changes use the full-subset scoring protocol; these denominators can differ. The primary 28-flag routing pool is reported separately. *DeepSeek-R1-Distill-Qwen-32B uses K=16. †Qwen3-30B-A3B uses the September 5, 2026 snapshot, with 16 flags and overall K=32 MV accuracy of 86.4%.

The primary Qwen3-32B rows in this matrix use a separate 24-flag evaluation. They are retained as pair-level evidence and are distinct from the 28-flag, 143-question routing analysis. The table also includes dataset-specific subsets that differ from the larger transfer pools in Appendix F.8. Comparisons should follow the stated subset and scoring protocol rather than combine these rows as if they came from one common pool.

B Experimental Protocol and State Construction

B.1 Extraction and descriptive state fitting

All shares are normalized by the number of valid extracted answers. The primary $K=32$ routing pool requires at least 20 valid letter outputs. The runner-up reliability analysis additionally requires at least two distinct answers, since a second-ranked answer must exist, and uses at least 8 valid letters at $K=16$, 20 at $K=32$, and 32 at $K=64$. These eligibility rules and differences between collection snapshots produce distinct analysis pools; in particular the runner-up pools are smaller than the routing pools they are drawn from, and the qualifying count in each row of Table 6 reflects this.

We fit the three-component Dirichlet–multinomial mixture by expectation–maximization on sorted count vectors. Majority share, top-two mass, and entropy summarize the fitted regions, and each question receives its maximum-posterior assignment. The likelihood is descriptive because sorting re-indexes answer positions by frequency. The three-component interpretation is retained for readability; the decrease in BIC through six components does not support a uniquely identified three-state taxonomy.

For free-form AIME responses, numeric answers are extracted from `\boxed` outputs. We represent each question by the counts of its three largest answer groups and a residual bucket, yielding four cells as in GPQA. Applying the frozen GPQA fit gives 5 stable, 25 contest-like, and 15 diffuse-like assignments. These labels describe the transformed count vectors and do not validate the GPQA action thresholds on AIME.

B.2 Generation and verification protocol

Sampling uses a single user-role message with no system prompt, rendered through each checkpoint's chat template. The template asks for explicit `[Step 1]`, `[Step 2]` numbering and closes with the line `The final answer is \boxed{LETTER}.` Reasoning checkpoints receive a reduced variant that drops the instruction to emit only the option letter. Decoding uses temperature 0.7 and top-p 0.95. Output length caps differ across checkpoints and are recorded per generation.

An output enters the counts only if the extracted answer is a single letter inside the question's option set. Extraction removes any reasoning block, reads `\boxed` content, and otherwise falls back to a trailing "answer", "therefore", or "thus" clause. Shares are computed over retained outputs, so format failures do not enter the state.

Verification presents the question, the candidate letters in descending frequency order, and an instruction to output only the chosen letter, under a system prompt that suppresses thinking. It uses $M=3$ draws at temperature 0.3. When a question has fewer distinct answers than the candidate-set size, the remaining option letters are appended so that the set has fixed size. A draw that yields no valid letter is scored as retaining MV, which keeps the comparison denominator fixed.

B.3 Discovery, test, and routing pools

A question-level hash divides the 198 GPQA questions into 98 discovery and 100 test questions. In the frozen validation, the mixture is fitted and verifiers are screened using discovery questions; test questions then receive frozen assignments. The reported test summaries retain the main qualitative patterns: a positive screened-verification estimate in contest, a negative reflection estimate in stable consensus, and a negative off-gate verification estimate in diffuse uncertainty. The reported 13-question verification subset gains 30.8 pp. It is distinct from the 67-question eligible test pool, whose count gate selects nine questions for the matched-budget comparison.

The full primary routing pool has 143 questions. Of its 28 flags with $\eta \geq 0.3$, 17 have $t_2 \geq 0.9$ and 11 have $t_2 < 0.9$. All verification allocators use the same Gemma3-27B protocol: three calls per selected question at temperature 0.3. Outputs are reused when comparing allocation rules. One scattered question has no valid letter in any verifier draw; its verification outcome is scored as retaining MV. Keeping this fallback fixed prevents format failure from changing the comparison denominator.

The deployed gate observes answer counts alone; verifier screening and subsequent verification have separate costs. A mixture assignment is not required to execute the policy.

B.4 Additional sampling and generation budgets

Figure 4 compares candidate availability across sample budgets and runner-up reliability at $K=64$. On 141 paired primary-sampler questions, additional samples often change whether the gold answer is present without changing the MV prediction. Regional MV accuracy changes by less than one percentage point under the reported halving or doubling of the sample budget.

The nested sampling comparison uses 118 questions eligible at both $K=32$ and $K=48$ within the same $K=64$ run. MV accuracy increases from 53.4% on the first 32 trajectories to 54.2% on the first 48, a gain of 0.8 pp. The independent-run budget comparison in Appendix F.8 uses a different construction and a 52.5% baseline at $K=32$. The two estimates should not be combined.

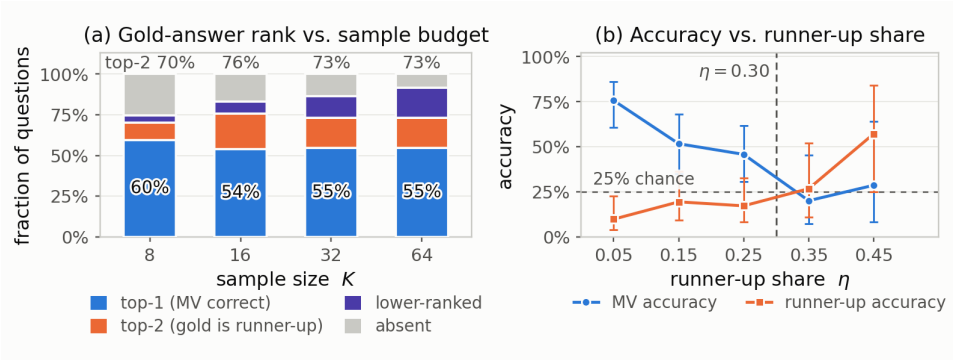

Figure 4: **Candidate availability can increase without resolving selection.** (a) Gold-answer rank across budgets on 141 paired Qwen3-32B questions. From $K=16$ onward, MV stays near 54–55%, while top-two coverage is about 73–76%. (b) MV and runner-up accuracy by η at $K=64$ ($n=129$), with Wilson 95% intervals. For $\eta < 0.3$, runner-up accuracy is 15.0%, below a uniform four-option reference (one-sided exact binomial $p=0.008$). That reference does not control for answer-position preferences.

B.5 Separating verification from abstention across samplers

Table 5 applies the same count thresholds to three samplers. Abstention increases accuracy among answered questions in all three settings, at reduced coverage. Verification adds positive point estimates for Qwen3-32B and InternLM3-8B, but a negative one for Qwen3-8B. The latter fails verifier screening, so its verify-and-abstain row is a forced-verification ablation, not the selected policy. Its selected policy abstains on scattered flags and otherwise keeps MV.

Outcome	Qwen3-32B	InternLM3-8B	Qwen3-8B
MV at full coverage	54.5%	35.9%	49.0%
Questions	143	181	192
Abstain on scattered: accuracy	56.8%	40.4%	50.6%
Coverage	92.3%	77.9%	84.4%
Also verify two-mode: accuracy	61.4%	41.8%	50.0%
Coverage	92.3%	77.9%	84.4%
Verification gain: answered set	+4.5	+1.4	−0.6
Verification gain: full pool	+4.2	+1.1	−0.5
Scalar-allocation full-pool gains (pp)			
Runner-up-share ranking	+2.1	+1.1	+1.0
Entropy ranking	+0.7	+0.0	−2.6
Random allocation	−1.3	+0.3	−0.7

Table 5: **Abstention and verification are separate sources of change.** Accuracy in selective rows uses answered questions. Verification-gain rows compare with abstention alone and use either the answered-set or full-pool denominator; gains are in pp. Qwen3-8B's verifier fails screening, so its selected policy uses abstention alone and the verification row is an ablation. Scalar allocation is reported on full pools and must not be compared directly with selective accuracy.

C Sampler-Specific Contestation Boundaries

The separate four-component model describes soft consensus, hard consensus, contest, and scattered responses. Its original diagnostic analysis uses a 70% training split, distinct from the question-hash discovery/test validation in Appendix B. It supplies $C(q)$ and is used to examine the count gate, not to replace the three-component descriptive summary or to execute routing. The September 5, 2026 fits give contest-component weights of 0.20, 0.29, 0.29, and 0.09 for Qwen3-32B, Qwen3-8B, Gemma3-27B, and Llama-4-Scout respectively. Transferring η=0.3 therefore tests a convenient proxy; it does not assume an optimal shared boundary.

For the primary sampler, discovery, test, and full-data refits place the $C(q)$=0.5 crossing at 0.311, 0.293, and 0.296. The learned gate overlaps with 9/13 count-rule flags on discovery, 12/15 on test, and 23/28 in the full pool. Refitting on test or full data is a sensitivity check, not frozen validation. Figure 5 illustrates the primary-sampler relationship.

Verifier judgment accuracy is approximately 71% on questions with $\eta \geq 0.3$, compared with 47% for $\eta \in [0.2, 0.3)$. Verification loses 11.5 pp in the latter interval. This result concerns a particular resolver and sample; a distributional boundary alone does not determine action value.

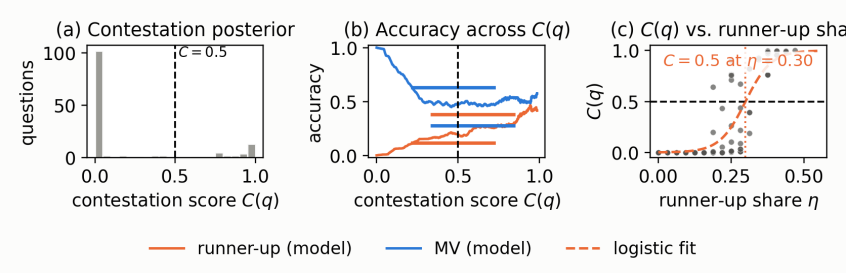

Figure 5: **A sampler-specific count proxy for contestation.** (a) Fitted $C(q)$ scores concentrate near zero and one. (b) Below versus above $C(q)$=0.5, runner-up accuracy changes from 11.5% to 37.9%, while MV accuracy changes from 62.8% to 27.6%; smoothed model predictions are also shown. (c) The logistic relationship with η crosses 0.5 near 0.30. This motivates an operational proxy on the primary sampler, not a universal uncertainty threshold.

D Runner-Up Reliability Across Configurations

Table 6 compares the correctness of the second-ranked answer below and above η=0.3. Of the six GPQA configurations, two retain a significant difference after Benjamini–Hochberg (BH) adjust-

ment across all seven table rows: Qwen3-32B at K=32 and Llama-4-Scout. Qwen3-32B at K=64 has unadjusted p=0.032 and adjusted p=0.075; the remaining GPQA rows are inconclusive.

GLM-4.7-Flash is excluded from this analysis because frequent truncation leaves only eight questions meeting the extraction criterion. Across the seven listed configurations, a question-clustered bootstrap estimates a 14.3 pp high-minus-low-η difference (95% CI $[7.1, 21.5]$). This pooled contrast is evidence about runner-up availability. It does not establish that verifying every large runner-up group improves accuracy.

Model / dataset	K	n	Runner-up acc. $\eta < 0.30$	Runner-up acc. $\eta \geq 0.30$	MV acc. $\eta \geq 0.30$	Fisher p	BH p
Qwen3-32B / GPQA	32	116	14.8%	42.9%	28.6%	.003	.022
Qwen3-32B / GPQA	64	129	15.0%	36.4%	22.7%	.032	.075
Llama-4-Scout / GPQA	32	88	16.2%	45.0%	35.0%	.013	.046
Qwen3-8B / GPQA	32	164	26.1%	26.4%	35.8%	1.000	1.000
Gemma3-27B / GPQA	32	121	26.3%	31.1%	42.2%	.676	.929
Qwen3-32B / MMLU-Pro	32	280	17.8%	31.6%	42.1%	.076	.133
Qwen3-32B / GPQA	16	110	23.8%	26.9%	30.8%	.796	.929

Table 6: **Runner-up reliability depends on the configuration.** Runner-up accuracy is the fraction whose second-ranked sampled answer is correct. Fisher's exact test compares the two η groups, and BH adjustment covers all seven rows. MV accuracy uses the same high-η subset. The count n is the analyzed pool for the row, not the primary routing denominator.

E Mechanism Checks and Alternative Signals

E.1 What the descriptive model can explain

Within the four-component contest model, an incorrect leading answer can retain more modeled mass than the correct runner-up. Sampling or re-examining the problem with the same source can therefore preserve its preference. This is a descriptive explanation consistent with the verification results, not an identified causal mechanism. Table 7 retains the model checks from their respective analysis snapshots. Their flag counts and fitted-null references should not be substituted for the primary routing-pool counts.

Check	Model output	Corresponding observation
Runner-up reliability	.07 / .20 / .27 / .33 / .67	.07 / .20 / .29 / .33 / .67
Low-η runner-up accuracy	.157	.161; fitted-null reference .263
Contestation transition	.39 / .66 / 1.0 over $\eta \in [.2, .5)$	Threshold near .30
Contest assignment among flags	.72 / .90 / .98 at K=16/32/64	Transition around K=32
Flag frequency	.062 / .12	.059 / .196 (28/143)
Same-source rescue	Below-chance prediction in the contest component	21.7% rescue rate
Contestation gate	$C(q) > .5$	29 flags; 21 also satisfy $\eta \geq .3$

Table 7: **Descriptive checks of the four-component model.** Entries retain the corresponding model-check analyses. Rescue conditions on recoverable MV errors; the fitted-null reference differs from the uniform-option reference. The reported 29-flag overlap is kept separate from the primary-pool 23/28 overlap in Appendix C. These checks describe fit and consistency, not independent validation of a failure taxonomy.

E.2 Behavioral confidence and net verification value

Behavioral confidence is the mean trajectory token log-probability; it is different from majority share and is not used by the deployed router. Splitting at its median, runner-up accuracy changes from 8.2% to 60.0% across η=0.3 in the high-confidence group (Fisher p=0.001), but from 17.5% to 21.4% in the low-confidence group (p=0.71). Top-two coverage on flags is 90% versus 43%, and verification gains are 20.0 pp on ten high-confidence flags and 14.3 pp on fourteen low-confidence flags. These exploratory patterns suggest a possible moderator without establishing a confidence-by-disagreement interaction. Verbalized confidence has little discrimination (AUC 0.52).

Net action value also depends on the number of correct baseline answers exposed to damage. In the matrix snapshot, MV is correct on 6/24 Qwen3-32B flags and 15/41 Qwen3-8B flags. Reported verifier judgment accuracies are 71.4% for Qwen3-32B, 57.1% for Llama-4-Scout, and 46.2% for Qwen3-8B. The primary and Phi-4 sampler cells gain 16.7 pp and 5.7 pp respectively. These quantities use different conditioning events, so verifier judgment accuracy should not be read directly as net gain.

Off-gate verification is particularly damaging on the primary sampler's diffuse, low-t_2 subset: it loses 29.2 pp on 24 paired questions (McNemar p=0.039). The eight flagged questions in that descriptive cell instead gain 12.5 pp, an imprecise estimate from a different subset. Neither cell should be conflated with the eleven scattered flags defined by the operational gate.

E.3 Other signals and sampling sensitivity

The tested trajectory-topology method identifies the runner-up less accurately than answer counts (18.9% versus 31.1%). Selecting by summed answer log-probability reduces primary-sample accuracy by 20.4 pp. Conditional resampling of a runner-up trajectory raises its answer share from 0.36 to 0.62 without increasing accuracy. Split-half consistency, trajectory length, and verbalized confidence provide little additional signal in the reported evaluations. These results favor count features among the tested alternatives, not a general claim that trajectory features are uninformative.

The reported sampling-noise threshold-exceedance probabilities fall from 47.5% at K=16 to 19.5% at K=32 and 4.4% at K=64. The runner-up transition becomes more pronounced near K=32, with a reported 20.8 pp contrast (p=0.008). This is a sample-size sensitivity result, not a universal minimum budget for diagnosis.

E.4 Option order and answer-space effects

An option-order control resamples the primary model at K=32 using a hash-seeded permutation for each question, then maps extracted letters to canonical option identities. On 141 questions eligible in both runs under that pipeline, mapped MV accuracy is 63.8% in each run. This subset and extraction pipeline differ from the 143-question routing pool. Unmapped letters match the canonical gold letter on 21.2%, which measures agreement between letter encodings rather than task accuracy.

After mapping, answer-letter marginals change from 31.9/21.3/28.0/18.8% to 25.6/24.7/26.0/23.5%. However, answer-format compliance changes from 93.8% to 77.6%, and 43 questions are excluded from the paired analysis. The original low-η effect is itself nonsignificant on this paired subset (p=0.256). The control therefore does not establish independence from option order.

Across seven evaluated configurations, the runner-up-minus-MV accuracy difference is negatively correlated with MV reliability ($r = -0.79$). Reported differences are 0 pp on MATH, -10 pp on MMLU-Pro, and -24 pp on ScienceQA. In a separate AIME K=64 analysis, only one of 38 questions has a large runner-up group. These results show that a useful alternative candidate need not be captured by the same count threshold across tasks.

E.5 Reflection as an action, not a default remedy

Across the reported targets, standard reflection corrects 19/163 recoverable failures (11.7%) and has a negative overall accuracy change. Multi-round reflection often switches answers, but its difference from a blind-switching baseline outside the flagged set is inconclusive (p=0.54). This does not establish equivalence. Table 8 reports paired flagged and unflagged outcomes separately. The routing implication is to measure the value of the specific reflection protocol instead of assuming that uncertainty makes reflection beneficial.

Sampler / dataset	Flagged n	MV→ reflection	Unflagged n	MV→ reflection
Qwen3-32B / GPQA	24	.250 → .250	118	.619 → .525
Qwen3-32B / ScienceQA	35	.543 → .429	451	.856 → .831
Qwen3-32B / MMLU-Pro	15	.733 → .467	238	.815 → .702
Qwen3-8B / GPQA	41	.366 → .195	110	.555 → .473
Gemma3-27B / GPQA	30	.400 → .433	79	.595 → .481
Llama-4-Scout / GPQA	18	.333 → .278	152	.658 → .625
GLM-4.7-Flash / GPQA	0	—	—	.974 → .816

Table 8: **Paired reflection outcomes on the reported subsets.** Accuracy is a proportion. Each arrow compares the same questions before and after reflection; flagged and unflagged subsets are separate. GLM-4.7-Flash has no flagged questions in this analysis, and its evaluated unflagged sample size is unavailable in the source summary.

F Statistical Evidence, Ablations, and Transfer

F.1 Intervals and inferential scope

Unless stated otherwise, confidence intervals use 10,000 question-level bootstrap resamples, preserving before/after pairing. Table 9 collects the principal estimates and their conditioning sets. For pooled configuration analyses that use a question-clustered bootstrap, the question is the resampling cluster.

The primary analyses concern same-source rescue, screened-verifier rescue on the standard GPQA targets, and equal-budget routing against random allocation. Subgroup, transfer, and threshold sweeps are exploratory. Table 6 adjusts its seven comparisons using BH. We distinguish bootstrap intervals, exact paired tests, and pool-randomization tests because they address different uncertainty summaries or null hypotheses. In particular, the primary routing gain's bootstrap interval excludes zero, while its exact paired comparison against MV gives p=0.070.

Statistic	Estimate	95% CI
Rescue rates conditioned on recoverable MV errors		
Same-source pooled (n=46)	21.7%	[10.9, 34.8]
Gemma3-27B, standard GPQA targets (n=27)	70.4%	[51.9, 85.2]
InternLM3-8B with Gemma3-27B (n=19)	73.7%	[52.6, 89.5]
DeepSeek-R1-Distill with Gemma3-27B (n=12)	50.0%	[25.0, 75.0]
Qwen3-30B-A3B with Gemma3-27B (n=8)	37.5%	[12.5, 75.0]
Qwen3-30B-A3B with Phi-4 (n=8)	87.5%	[62.5, 100]
Paired action changes within descriptive regions (pp)		
Stable: reflection (n=44)	−9.1	[−20.5, +2.3]
Contest: reflection (n=67)	−9.0	[−17.9, +0.0]
Contest: same-source verification (n=13)	−15.4	[−38.5, +0.0]
Contest: screened verification on flags (n=13)	+30.8	[+0.0, +61.5]
Diffuse, low t_2: off-gate verification (n=24)	−29.2	[−50.0, −8.3]
Paired verification changes on selected subsets (pp)		
Two-mode gate (n=17)	+35.3	[+5.9, +64.7]
Reported frozen-model test subset (n=13)	+30.8	[+7.7, +53.8]
Three screened samplers pooled (n=75)	+10.7	[+0.0, +32.0]
Equal-budget full-pool changes: 143 questions, 51 calls (pp)		
Failure-aware gate	+4.2	[+0.7, +7.7]
Runner-up-share ranking	+2.1	[−2.1, +6.3]
Entropy ranking	+0.7	[−2.8, +4.2]
MV-confidence ranking	+0.0	[−3.5, +4.2]

Table 9: **Bootstrap uncertainty and analysis denominators.** Rescue rates and paired gains use different conditioning sets. The two n=13 rows represent distinct analyses. An interval above zero for one allocator does not establish a difference from another allocator. The gate's exact paired comparison against MV gives p=0.070, despite its positive bootstrap interval.

F.2 Threshold sensitivity and scalar allocation

The learned contestation crossing motivates η=0.3; Appendix C reports its refit sensitivity. The t_2=0.9 cutoff separates concentrated from scattered flags. A retrospective sweep yields positive full-pool point estimates at every listed cutoff (Table 10), but changes both the selected subset and

the number of calls. Discovery verification outcomes do not distinguish cutoffs from 0.8 to 0.95. Thus, the sweep assesses local sensitivity rather than independently validating 0.9 as optimal.

Minimum t_2	Verified	Scattered	MV→ ver.	Selected Δ	Rescue	Damage	Pool Δ
.80	25	3	.320 → .520	+20.0	8	3	+3.5
.85	23	5	.348 → .522	+17.4	7	3	+2.8
.875	21	7	.286 → .524	+23.8	7	2	+3.5
.90	**17**	**11**	.294 → .647	**+35.3**	7	1	**+4.2**
.925	16	12	.312 → .625	+31.2	6	1	+3.5
.95	16	12	.312 → .625	+31.2	6	1	+3.5

Table 10: **Retrospective sensitivity to top-two concentration.** Every rule starts from the same 28 questions with $\eta \geq 0.3$ and uses three Gemma3-27B calls per verified question. Changes are in pp. Full-pool gains use all 143 questions and retain MV on unverified questions. The scattered count indicates potential abstentions in the complete policy. The sweep changes both selection and call budget and does not establish an optimal threshold.

Pool randomization tests exchangeability of per-question gains. It favors the count gate over entropy (p=0.008), but not runner-up-share allocation (p=0.11). Exact paired tests instead compare the fixed policies' final answers on the same questions; neither comparison establishes superiority. The direct gate-versus-η comparison gives p=0.375 on the full pool. The random-allocation reference in Figure 2 is p=0.0002 for the probability of a uniformly selected equal-budget allocation matching or exceeding the reported gate gain.

We also select scalar-only quantile intervals on discovery questions and freeze their half-open boundaries for testing (Table 11). None exceeds its corresponding monotone ranking in test point estimate, although budgets differ. The matched-budget comparison uses 27 calls and ties at 40/67 correct for the gate and top-η, from 36/67 for MV. It is the relevant test of accuracy at the gate's realized test budget; the other table blocks answer different budget questions.

Test policy (n=67)	Verified	Calls	Rescues	Damages	Test-pool Δ
Intervals selected on discovery questions					
$\eta \in [4/15, 5/14)$	8	24	0	0	+0.0
$H \in [.8444, .9032)$	5	15	0	2	−3.0
$p_1 \in [13/21, 2/3)$	4	12	0	0	+0.0
Margin $\in [3/14, 1/3)$	5	15	1	1	+0.0
Monotone ranking: 51 calls					
Highest 17 η	17	51	5	2	+4.5
Highest 17 H	17	51	3	3	+0.0
Lowest 17 p_1	17	51	5	3	+3.0
Lowest 17 margins	17	51	5	3	+3.0
Matched budget: 27 calls					
Failure-aware gate	9	27	4	0	+6.0
Highest 9 η	9	27	5	1	+6.0
Lowest 9 p_1	9	27	3	2	+1.5
Lowest 9 margins	9	27	3	2	+1.5

Table 11: **Frozen scalar rules and matched-budget testing.** Changes are in pp over the 67-question test pool with 36 correct MV answers. Margin is $p_1 - \eta$. Interval boundaries are selected on discovery questions and then frozen. The final block matches the count gate's realized 27-call budget and shows equal accuracy for the gate and top-η. The first two blocks use different budgets.

F.3 Verifier selection and denominator conventions

Four verifier candidates are evaluated for the primary sampler, and Gemma3-27B is selected using discovery questions. Table 12 describes the candidates over the full 28-flag analysis; it is not an independent evaluation of selection. Gemma3-27B corrects 9/19 MV errors with valid outputs. Conditioning on recoverability changes that fraction to 9/11. Its full-flag net gain is $6/28$=21.4 pp, or $6/143$=4.2 pp over the full routing pool.

Screening is also stable with respect to the split itself: repeating the discovery/test hash split under 20 seeds selects Gemma3-27B in 18 of them, and the selected verifier's test-half full-pool gain is non-negative in every seed (mean +3.6 pp, range $[0.0, 9.2]$).

The reported frozen-model verification subset of 13 questions gains 30.8 pp with bootstrap interval $[7.7, 53.8]$. The separate 13-question descriptive-contest cell in Table 2 has the same point estimate but interval $[0.0, 61.5]$. The subsets and intervals are distinct. The matched-budget eligible test comparison in Table 11 is a third analysis and verifies nine questions.

Verifier	Corrected / MV errors with valid outputs	Flagged-set gain
Gemma3-27B	9/19	$+21.4$ pp
Qwen3-8B	4/18	$+14.3$ pp
GLM-4.7-Flash	5/13	$+10.7$ pp
Qwen3-32B (same source)	3/18	$+3.6$ pp

Table 12: **Candidate verifiers on the primary 28-flag analysis.** Correction fractions condition on valid outputs; net gains divide rescues minus damages by all 28 flags. These full-pool summaries describe candidate performance after discovery-based selection and are not an independent test of the selection procedure.

F.4 Mixture granularity and pooled verification

BIC improves throughout $k{=}2, \ldots, 6$. Two components merge regions separated by the three-component fit; larger fits subdivide them. The chosen granularity is interpretive, and the deployed thresholds do not require three-component assignments.

Unscreened cross-family verification gains 2.3 pp over 307 flagged evaluations (permutation $p{=}0.088$); same-source verification loses 5.6 pp over 126 ($p{=}0.336$). The screened pool gains 10.7 pp over 75 ($p{=}0.023$). Because these aggregates contain different model pairs and questions, their difference does not estimate the causal effect of screening. They motivate measuring correction and damage for each intended pair.

F.5 Structure and candidate availability

The regional statistics in Table 1 describe average headroom. The same separation appears at the level of individual flags. Among flags with $\eta \geq 0.3$ whose MV answer is wrong, the gold answer is the runner-up, already available among the top two candidates, on 50/78 (64.1%) of two-mode questions and 36/90 (40.0%) of scattered ones (Fisher exact odds ratio 2.68, $p{=}0.0021$). All seven samplers with such flags show the same direction. Because top-two concentration is correlated with the runner-up share, we also condition on η: in a regression of recoverability on η and a two-mode indicator, the indicator stays positive ($\hat{\beta}{=}0.23$, $t{=}2.71$, $p{=}0.0074$, $n{=}168$), and a Mantel–Haenszel test stratified by η gives a pooled odds ratio of 2.58 ($p{=}0.0045$). At the same runner-up share, concentrated contests roughly double the odds that a correct candidate is already present.

This concerns candidate availability, not realized accuracy. Converting availability into accuracy additionally requires a resolver whose net effect is positive for that sampler. On the primary sampler, η is associated with per-question verification gain ($p{=}0.0004$), but a two-mode indicator or t_2 adds no significant incremental association; within the 28 flags the two-mode coefficient is positive but inconclusive ($\hat{\beta}{=}0.37$, $p{=}0.17$). Extending the pipeline to three further pools (MMLU-Pro, ScienceQA, MATH) preserves the same division. Availability separates sharply: 62.5% versus 40.5%, odds ratio 2.44, $p{=}0.0013$, and $p{=}0.034$ after conditioning on η. The resolver's accuracy given that a correct candidate is present is the same in both arms (0.61 versus 0.59). The binding constraint is the resolver, not the structure: even the screened verifier selects the correct candidate from an available pair only about three times in five. The 35.3 pp versus 0 pp contrast accordingly remains a primary-pair result, and the gain an action realizes depends on the verifier–sampler pair, as the matrix in Appendix A shows.

F.6 A worked routing example

For one GPQA organic-chemistry question, 28 of 32 trajectories yield valid letters: 15 select D and 13 select C. MV returns D, but C is correct. The observed state gives $\eta{=}13/28{=}0.464$ and $t_2{=}1.0$, so the rule invokes verification. The screened verifier selects C in all three draws, correcting

the answer with three additional calls. The gold answer is used here to explain the outcome; it is unavailable to the count rule at deployment.

F.7 Accuracy and resource accounting

Figure 6 uses 118 questions eligible at each $K \in \{16, 32, 48, 64\}$. Taking each budget from its own run yields MV accuracies of 52.5%, 52.5%, 54.2%, and 53.4%. Starting from $K{=}32$, the structure gate verifies 13 questions with 39 calls, adding $39/118{=}0.33$ generations per question. Accuracy rises from 62/118 to 69/118 (58.5%), a 5.9 pp full-coverage gain. Adding abstention yields 61.5% selective accuracy at 92.4% coverage. These denominators differ from the primary 143-question pool.

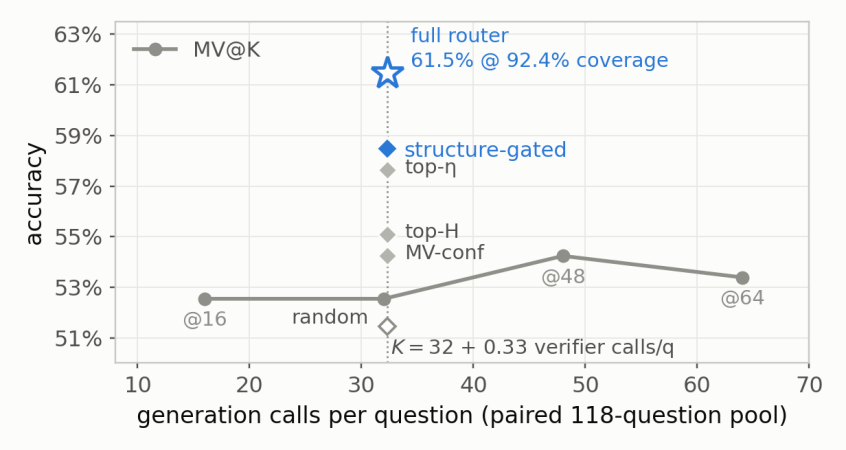

Figure 6: **Generation-call accounting on the fully paired 118-question pool.** Independent-run MV changes little over the plotted sampling budgets. Near $K{=}32$, 39 targeted verifier calls add 5.9 pp at full coverage. The open star additionally abstains, reaching 61.5% accuracy at 92.4% coverage. Calls are the common accounting unit; they do not equalize model FLOPs, latency, or token consumption.

In the measured output-token sample, a primary-sampler trajectory generates approximately 969 tokens and a fresh-solver trajectory 679. Each of 36 verifier draws generates three tokens. One verifier call thus has about 323 times fewer output tokens than one sampler trajectory. Three verifier calls generate nine tokens, compared with about 21,700 for a 32-trajectory fresh solve, a ratio near 2,400. These observations concern output length only: input tokens, model size, and runtime are excluded. They must not be presented as measured end-to-end compute savings.

F.8 Cross-dataset routing, fresh generation, and wider candidates

Cross-dataset routing. Without dataset-specific verifier screening, gains are inconsistent. The MMLU-Pro pool contains 435 eligible questions with 72.4% MV accuracy. Its 38 flags comprise 23 two-mode and 15 scattered questions. Verification has three rescues and three damages; adding abstention yields 74.3% accuracy at 96.6% coverage. On 500 ScienceQA questions, MV accuracy is 82.6%. Verification on 34 two-mode flags has six rescues and eight damages, losing 0.4 pp over the pool. The complete policy reaches 82.3% accuracy at 99.4% coverage. These evaluations do not support uniform transfer of either verification or the complete policy.

Fresh generation. Table 13 replaces the original answer with another model's $K{=}32$ MV answer on the 28 primary flags. Phi-4 answers 11/17 two-mode questions correctly, matching the screened verifier, and 4/11 scattered questions correctly, one more than MV or verification. This one-question difference is insufficient to establish an exploration benefit. Gemma3-27B as a fresh solver answers 8/17 and 2/11 respectively. On two-mode questions, verification exceeds Gemma3-27B fresh solving by three correct answers, but the paired difference is inconclusive (four favorable and one unfavorable discordant pair; exact $p{=}0.375$). Fresh solving uses a different call and token budget from verification.

Flagged subset	MV, K=32	Gemma3-27B verify, M=3	Phi-4 fresh, K=32	Gemma3-27B fresh, K=32
Two-mode (n=17)	29.4%	64.7% (+35.3)	64.7% (+35.3)	47.1% (+17.6)
Scattered (n=11)	27.3%	27.3% (+0.0)	36.4% (+9.1)	18.2% (−9.1)

Table 13: **Selection and fresh-generation probes use different budgets.** All 28 questions satisfy $\eta \geq 0.3$; t_2=0.9 separates the subsets. Parentheses give paired changes from MV in pp. Fresh generation uses 32 trajectories and MV, while verification uses three calls restricted to two candidates. These outcomes do not constitute equal-cost policy comparisons.

Wider candidate lists. Among scattered flags, the sampled-support oracle includes the gold answer on 6/11 questions at top two and 11/11 at top four. A separate expanded-option verifier pads candidate lists with zero-vote options when necessary; at top four, the list is the complete original option set and includes the gold answer on all 28 questions. This padded-list coverage differs from sampled-support coverage.

Gemma3-27B's correct counts at top two, top three, and top four are 14, 9, and 11 overall; 11, 8, and 8 on two-mode flags; and 3, 1, and 3 on scattered flags. Increasing candidate availability does not reliably increase realized accuracy under this verifier and prompt. The results reinforce the separation between an available correct answer and a competent resolver.

F.9 Free-form answers and missed dispersed failures

The count representation does not require option letters, but a bounded multiple-choice set can concentrate incorrect responses. We therefore examine canonical numeric answers extracted from boxed outputs with the primary sampler at K=32.

GSM8K. MV accuracy is 98.8% on 80 questions. Only one question triggers the gate: its 32 answers split between two numeric values, with the gold answer second (η=0.31, t_2=1.0). The unscreened numeric verifier selects it in all three draws. This case demonstrates that the configuration can occur in free-form responses; it provides no reliable estimate of policy-level transfer.

AIME. On the 45-problem set from the 2024 I/II and 2025 exams, median and maximum runner-up shares are 0.10 and 0.21, compared with 0.14 and 0.46 on the primary GPQA pool. No question satisfies $\eta \geq 0.3$, compared with 19.6% on GPQA. The median number of distinct answers is twelve versus two. Although forty AIME questions receive contest-like or diffuse-like mixture assignments, the deployed gate detects none.

The mismatch is substantive. MV accuracy is 40.0%; the gold answer is below rank two on another 22.2% and is never sampled on 37.8%. The current rule retains MV in both error cases, and perfect top-two selection cannot help. The first case requires considering candidates beyond the leading pair; the second requires generating a missing candidate. Whether either intervention has positive net value remains to be tested. A small runner-up group must therefore not be treated as a general certificate of stable consensus.