RETHINKING LLM PRUNING VIA TEST-TIME SCALING

Anonymous authors
Paper under double-blind review

ABSTRACT

Model pruning aims to compress large language models while preserving their capabilities, yet its effects on reasoning remain incompletely understood. We revisit expert, FFN and layer pruning through test-time scaling on mathematical reasoning and coding benchmarks. Pass@k curves distinguish success frequency from finite-budget reasoning coverage. Pruned models can preserve or exceed the original model's coverage even when sampling efficiency declines. Expert pruning more consistently preserves accuracy and coverage than FFN or layer pruning in the core matched-rate comparisons. Expert selection further affects both coverage and the sampling and token costs of reaching it, with advantages that vary across backbones. As removal increases, retained coverage can require substantially greater generated-token work, while severe pruning can also degrade generation stability. A mean-accuracy-matched uniform-loss reference does not fully explain the observed coverage losses. Models pruned from larger backbones can also retain broader reasoning coverage than native smaller models, although these comparisons do not match total parameters or training compute. Together, these findings provide an empirical guide to pruning that considers reasoning coverage, sampling efficiency and generation stability jointly.

1 INTRODUCTION

Structured pruning removes experts, intermediate channels, or layers to reduce model size (Lu et al., 2024; Ma et al., 2023; Men et al., 2025). What happens to the model's reasoning capabilities when these structures are removed? Pruning may make successful solutions less likely, eliminate access to previously solved problems, or change which problems can be solved. Distinguishing these possibilities is necessary to understand what compression preserves and what it changes.

We study this question through repeated sampling (Chen et al., 2021; Brown et al., 2024). By following pass@k from one to many candidates, we compare how readily a model succeeds with the range of problems on which it produces a successful candidate. Large-budget pass@k has been used to study reasoning after reinforcement learning (Yue et al., 2025), while recent work examines test-time scaling after layer pruning (Wang et al., 2026). Here, test-time scaling is a lens on the effects of pruning, not a recovery or budget-allocation method.

We compare expert, FFN and layer pruning on Qwen3.6-35B-A3B and GLM-4.7-Flash across two mathematics and two code tasks, with Gemma-4-26B-A4B-it and four dense backbones extending the study. The comparisons yield six findings about reasoning after pruning, developed in Sections 3–6.

♦ **Pruning can broaden reasoning coverage beyond the original model.** Comparing pass@k curves with the unpruned parent, some children start below it at small k and cross above it as k grows, so lower single-sample accuracy need not mean a narrower reachable set. Extending the budget on the slowest task keeps adding solved problems (Sections 3.1 and 3.2).

♦ **Expert pruning more consistently preserves single-sample accuracy than FFN or layer pruning.** At a shared nominal removal rate on the core grid, expert removal keeps both single-sample accuracy and the pass@k endpoint closer to the parent than width or depth removal, which makes it the reasonable default at moderate rates. The ordering weakens under severe removal (Section 3.2).

♦ **Expert selection affects both reasoning coverage and sampling efficiency.** At fixed removal, criteria can reach the same coverage with different success frequencies and output lengths. On another

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.

backbone, even their coverage endpoints separate. Selecting experts therefore changes how readily retained reasoning is accessed, not just how many experts remain (Section 3.3).

♦ **Preserved reasoning coverage can coexist with reduced token efficiency and degraded generation stability.** Expert pruning can leave output lengths nearly unchanged yet require many more attempts to reach a target coverage. Severe FFN and layer removal also change termination and repetition, so additional generation need not recover successful reasoning. These are distinct costs of compression, even when a coverage endpoint looks favorable (Section 4).

♦ **A uniform decline in success probabilities does not fully explain pruning-induced coverage loss.** Matching the child's mean accuracy by uniformly thinning parent successes still predicts more coverage than several pruned models attain on mathematics and code. A shared reduction in success frequency therefore misses changes in how success is distributed across problems (Section 6.1).

♦ **Models pruned from larger models can achieve broader reasoning coverage than native smaller models.** Light pruning retains an advantage over a native smaller model on both mathematics and code. More severe, width-matched comparisons can reverse that advantage. The relevant question is how much of the larger source's reasoning survives compression, rather than whether the child improves on its own parent (Section 5).

2 Experimental Setup

2.1 Pruning Interventions and Comparison Design

The core grid pairs 48 Qwen3.6-35B-A3B and GLM-4.7-Flash children with their parents. Gemma-4-26B-A4B-it and four dense backbones extend the cohort (Appendices B.3 and B.4). Every Qwen3.6-35B-A3B result below uses the post-trained release.

We use RAZOR for routed experts (Song & Zheng, 2026), Taylor saliency for FFN channels without recovery training (Ma et al., 2023), and a perplexity-greedy depth selector adapted from iterative block removal (Song et al., 2024). Tail-drop and alternative expert criteria are controls. Removal rates count structural units, not matched parameters or FLOPs, and the depth sweeps have different endpoints (Appendices A.1 and A.3).

2.2 Test-Time Scaling and Evaluation Protocol

For problem q, let c_q of n_q generated completions from checkpoint m be correct under the benchmark's grader. For $1 \le k \le \min_q n_q$, we use the estimator of Chen et al. (2021)

$$P_m(k) = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \left[1 - \frac{\binom{n_q - c_q}{k}}{\binom{n_q}{k}} \right]. \tag{1}$$

$P_m(1)$ is single-sample accuracy, while larger k measures the fraction of problems reached by repeated sampling. At common K, it is observed coverage when all $n_q = K$ and otherwise the coverage estimate for a uniformly drawn K-candidate subset. Endpoints remain conditional on the finite pool, prompt, decoder, length cap and grader. Zero hits do not imply zero success probability.

Relative to the same backbone's unpruned parent, denoted by 0, define

$$E_m = \frac{P_m(1)}{P_0(1)}, \qquad R_{m,K} = \frac{P_m(K)}{P_0(K)}. \tag{2}$$

These are pass@1 and fixed-budget coverage retention, not inference-cost measures, and either can exceed one.

All tasks use $T = 0.7$ and an 8,192-new-token cap. Nucleus sampling uses $p = 0.95$ and decoding top-$k = 20$ (Holtzman et al., 2020). The common mathematics pool is $n = 2048$ per problem, with selected Minerva (Lewkowycz et al., 2022) checkpoints extended to $n = 4096$. HumanEval+ (Liu et al., 2023) uses $n = 512$, and LiveCodeBench (Jain et al., 2025) is compared at common $K = 256$. Curves never extrapolate beyond recorded pools. Gemma-4-26B-A4B-it's corrected 272-problem Minerva set is not absolutely comparable with the 219-problem sets used by other backbones. Appendix A specifies prompts, subsets, grading, unequal pools and unrecorded settings. These are fixed-protocol curves, not temperature-optimized envelopes.

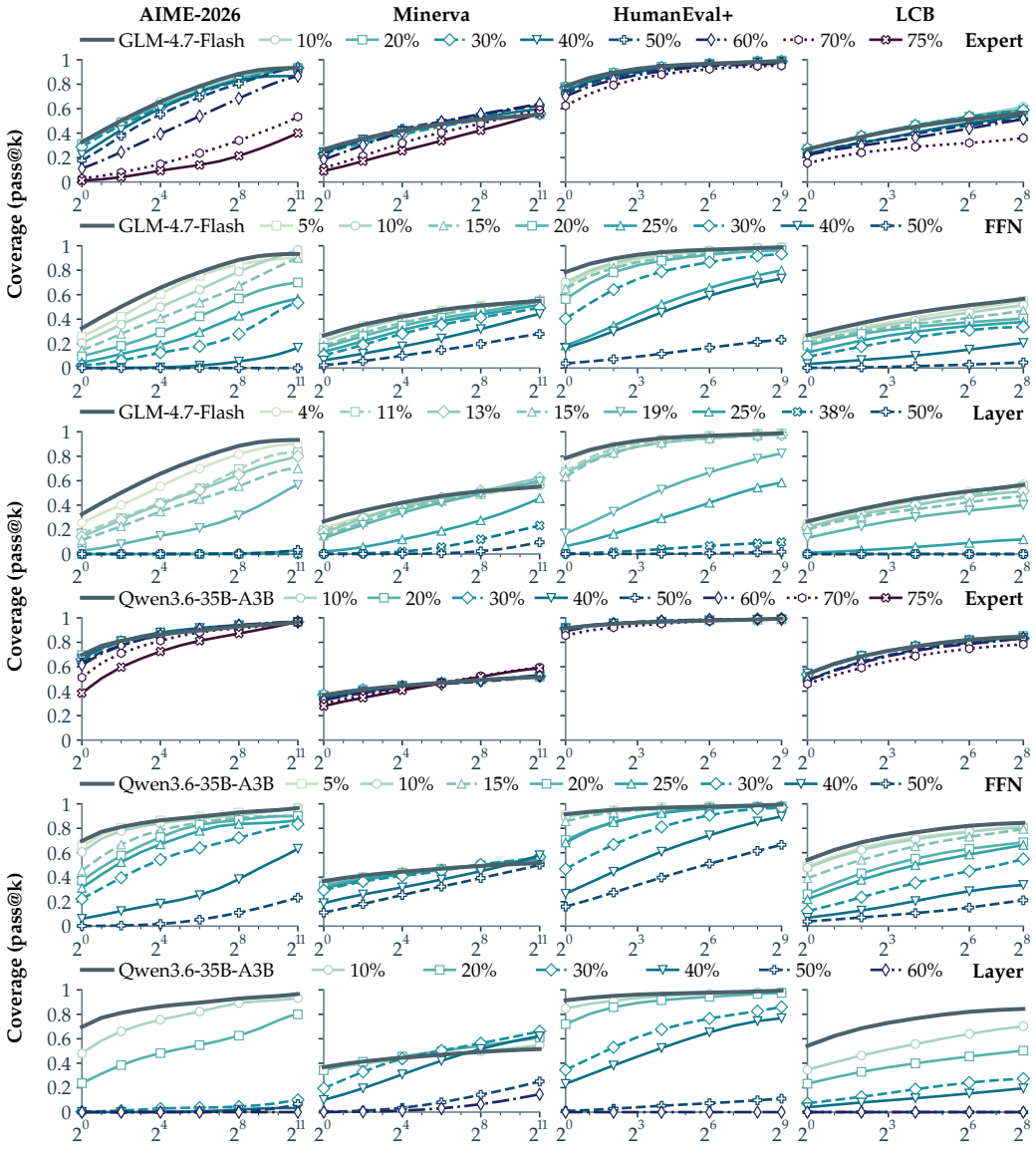


Figure 1: **Pruning sweeps on AIME-2026, Minerva, HumanEval+ and LiveCodeBench.** Rows are GLM-4.7-Flash then Qwen3.6-35B-A3B, one pruning axis each against its own parent, and columns are tasks. Mathematics stops at $k = 2048$ while code panels stop at their observed pools, so aligned panels need not share budget endpoints or saturation (Section 2.2). Expert-25% is omitted from display only, and expert-75% is unavailable rather than zero on code.

3 How Pruning Changes Reasoning

Figure 1 follows reasoning performance from one candidate to many across pruning structures and removal rates. Reading along each curve shows whether repeated sampling compensates for lower success frequency. Comparing curves across axes shows which structures preserve coverage more consistently. We first examine these two patterns, then ask how they depend on the task and on the choice of experts to remove.

3.1 Preserved and Expanded Reasoning Coverage

Obs. 1. At finite budgets, pruning can change success frequency without shrinking the sampled solvable range, and can sometimes expand it. On AIME, Qwen3.6-35B-A3B expert-75% lowers

pass@1 from 69.6% to 38.5% but matches its parent's 29/30 solved problems at $K = 2048$. In the common 2,048-candidate Minerva pools, the same model reduces pass@1 from 36.8% to 27.8% while increasing coverage from 113 to 129 of 219 problems. On LiveCodeBench at common $K = 256$, 10% expert removal raises coverage from 56.6% to 61.7% for GLM-4.7-Flash and from 84.4% to 85.7% for Qwen3.6-35B-A3B (Figure 1). Pass@1 also increases slightly in these two comparisons.

Analysis. **Pruning can make successful reasoning less frequent without making it less broadly accessible.** At $K = n$, each problem with a recorded success contributes equally to coverage, whereas pass@1 reflects how often it succeeds. A model can therefore lose repeated successes on some problems while reaching new ones. The Minerva and LiveCodeBench gains place this possibility in both mathematics and code. Extended Minerva sampling still adds solved problems, so the measured endpoints need not be saturated (Appendix D.4).

3.2 Effects of Pruning Structure and Severity

Obs. 2. Expert pruning preserves reasoning more consistently than FFN or layer pruning in the core matched-rate comparisons, but severe removal exposes differences between backbones. At the shared nominal 50% removal rate, expert pruning attains higher coverage than FFN and layer pruning on all four tasks for both core backbones (Figure 1) and more consistently preserves single-sample accuracy. Severity sweeps expose the boundary. On AIME, Qwen3.6-35B-A3B still solves 29/30 problems after 70% expert removal, whereas GLM-4.7-Flash solves 16/30, and at 50% FFN removal Qwen3.6-35B-A3B solves only seven. Gemma-4-26B-A4B-it likewise loses substantial coverage under severe expert removal (Figure 5).

Analysis. **The pruning axis changes how much removal reasoning can tolerate.** Expert removal offers the strongest starting point in the core grid because its advantage spans tasks at the shared rate. The severe-removal comparisons show that choosing an axis and choosing its removal rate are separate decisions. These comparisons match structural rates, not resources, and do not establish a common safe ratio across backbones.

Obs. 3. A pruning profile that preserves or expands coverage on one task need not transfer to another task or backbone. Several Qwen3.6-35B-A3B FFN and layer checkpoints exceed the parent's Minerva coverage despite lower pass@1, but deteriorate on AIME. Its layer-60% checkpoint retains nonzero Minerva coverage while solving no problems on AIME or either code task in the recorded pools. Gemma-4-26B-A4B-it shows the same task split under layer pruning (Figures 5 and 6 and Appendix B.3). Among dense models, a consistent structural ranking need not imply gains across tasks. At 25% removal, FFN pruning retains more coverage than layer pruning across both dense backbones and all four tasks, yet Qwen3.5-4B FFN-25% improves Minerva while degrading AIME and code (Appendix B.4).

A joint screen changes which checkpoints survive. Of 100 eligible MoE and dense checkpoints, 49 retain at least 90% of parent coverage on HumanEval+ at $K = 64$, but only 19 do so on AIME, HumanEval+ and LiveCodeBench simultaneously. Thus, a favorable single-task endpoint admits 30 configurations that the joint requirement rejects. These descriptive counts compare the same eligible set, not population failure probabilities (Appendix B.6.2).

Analysis. **The useful pruning range is the intersection of task-specific ranges.** A ratio that benefits one task can already damage another, even within code. The cross-task screen therefore changes the choice of checkpoint, not merely how cautiously its score should be described.

3.3 Expert Selection at Matched Removal

The structural comparison favors expert pruning, but does not tell us whether its robustness follows from the pruning axis or the choice of experts. Using the primary-sweep RAZOR checkpoints as a reference, we compare selection criteria at fixed backbone, task, removal rate and budget.

Obs. 4. At a fixed removal rate, the selection criterion can change both low-budget efficiency and common-budget coverage, and its advantage need not transfer across backbones. Figures 2 and 14 compare Qwen3.6-35B-A3B at 50% and 25% expert removal. At 50%, all four criteria reach 96.7% AIME coverage at $k = 2048$, although their curves differ at smaller budgets. On this same task, FREQ and EAN average 4,761 and 5,327 tokens per output, so equal endpoint coverage need not

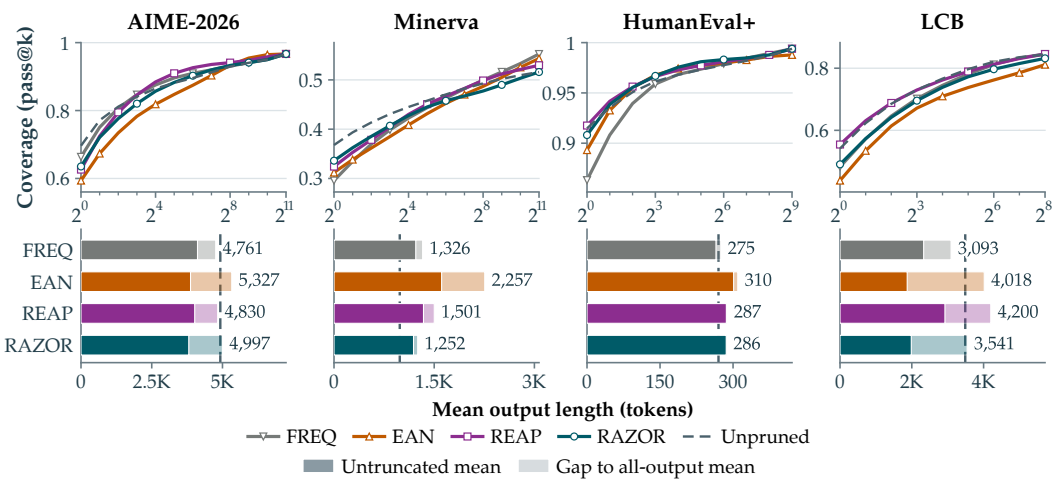


Figure 2: **Expert selection at 50% removal on Qwen3.6-35B-A3B.** The top row shows pass@k and the bottom row mean output length, where solid bars are conditional untruncated means and pale extensions reach the all-output means, including capped outputs. Dashed curves and vertical lines mark the parent. LiveCodeBench uses $n = 512$ for the parent and RAZOR and $n = 256$ for the other selections, so these panels should be read at common budgets.

require equal generation work even at the same candidate count. Selection also changes termination. On Minerva, RAZOR and EAN average 1,252 and 2,257 tokens per output, with truncation rates of 0.9% and 9.8%, respectively.

Nor is endpoint agreement intrinsic to expert pruning. On Gemma-4-26B-A4B-it at 50% removal and common $k = 256$, REAP reaches 40.0% AIME coverage versus 10.5% for RAZOR, and 59.4% versus 46.1% on the corrected 272-problem Minerva set (Appendix B.3). Depth-selection controls likewise show that the identities of removed structures matter beyond their number (Appendix B.5.2).

Analysis. **Selection determines more than the coverage endpoint.** When criteria reach the same endpoint, their success frequencies determine how many attempts are needed and their output lengths determine the work per attempt. The Gemma-4-26B-A4B-it comparison shows that the endpoint itself can also change. The next question is how these differences evolve as the removal rate increases, beyond the fixed-rate comparisons here.

4 Token Efficiency and Generation Stability

Coverage records whether successful answers remain accessible, not how much generation is needed to obtain them. Pruning can increase this work by requiring more attempts, making each attempt longer, or both. We first separate these contributions, then use Figure 3 to examine how generation changes across structures and removal rates.

4.1 Token Use and Efficiency

Obs. 5. Retaining reasoning coverage need not preserve sampling efficiency, and longer generation need not recover lost coverage. On AIME, Qwen3.6-35B-A3B retains its parent's 29/30 coverage at $K = 2048$ after both 50% and 75% expert removal. Reaching 90% of the parent's final coverage, however, requires 22 candidates at 50% removal and 246 at 75%, compared with 20 for the parent. The corresponding token work rises from 1.12 to 12.51 times the parent's work. Even at 75% removal, mean output length changes little, from 4,923.5 to 5,008.3 tokens. The additional work therefore comes primarily from needing more attempts, not longer ones (Appendices B.8.1 and B.9).

More attempts and longer attempts have different implications. For a fixed coverage target, generated-token work is $k^*_m \bar{t}_m$, where k^*_m is the smallest attaining candidate count and $\bar{t}_m$ is the

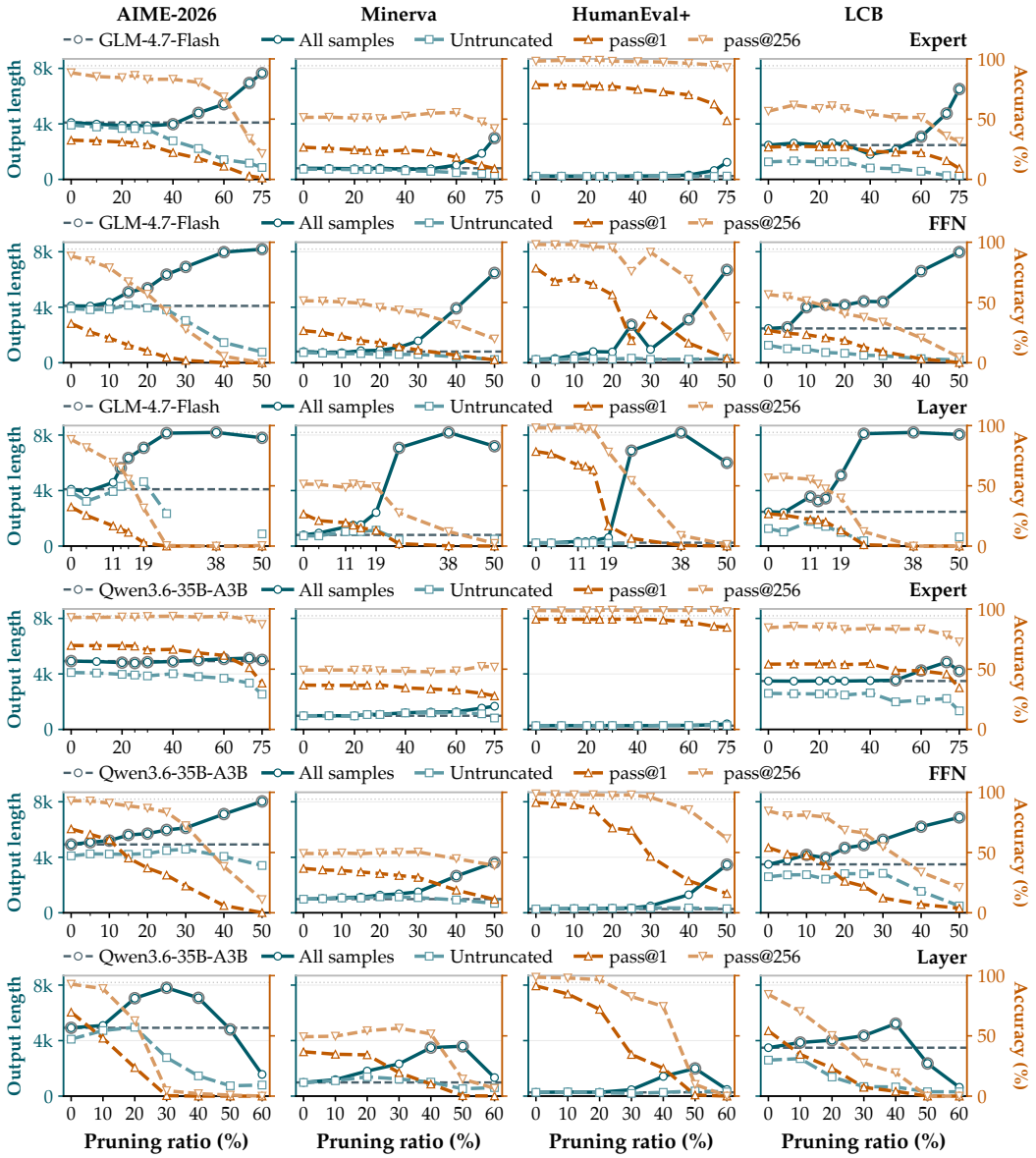


Figure 3: **Output length and accuracy against pruning ratio.** Layout follows Figure 1. The left axis gives mean tokens over all samples or over untruncated outputs only (k=1,000), with the dotted line at the 8,192-token cap, and the right axis gives pass@1 and pass@256 in per cent. The grey rule marks the parent's all-sample mean and an open grey ring marks truncation of at least 20%. Rows end at their own largest rate, not a shared removal fraction.

all-output mean length. The expert example separates retaining coverage from reaching it efficiently. Severe FFN removal presents a different pattern. All-output mean length rises while pass@1 and pass@256 fall across both core backbones and mathematics and code tasks (Figure 3). Thus, neither unchanged nor increased output length is sufficient to judge what reasoning survives. We next separate completed outputs from those that reach the length cap. Token work excludes prompt processing and verification, so it is not a measure of hardware efficiency.

4.2 Generation Stability

Separating longer outputs from more truncation. An increase in mean length can reflect longer completed outputs or more outputs reaching the cap. Figure 3 distinguishes these by showing both

all-sample and untruncated means. Within one output pool, let τ be the truncated fraction and $\bar{t}_{\mathrm{u}}$ and $\bar{t}_{\mathrm{c}}$ the means for untruncated and truncated outputs. Then $\bar{t}_{\mathrm{all}} - \bar{t}_{\mathrm{u}} = \tau(\bar{t}_{\mathrm{c}} - \bar{t}_{\mathrm{u}})$. The gap depends on both truncation and the lengths of completed outputs. It can remain small even with frequent truncation if completed outputs also approach the cap.

The divergence is informative when read alongside coverage, not as a standalone rule for choosing the pruning ratio. Figures 5 and 6 show whether coverage remains intact in the same range and how the answer changes with the sampling budget. A range that looks stable on one task may already lose coverage on another. The 20% markers in Figure 3 help locate increased truncation, but are display thresholds rather than fitted change points.

Termination alone is insufficient. In the 128-candidate trajectory pools, three of twelve backbone, axis and mathematics-task combinations show truncation falling after its peak while complete-trace coverage remains low. All three are layer sweeps. Shorter incorrect traces and greater repetition at severe removal provide further evidence that generation can stop without recovering successful reasoning (Appendices B.7 and C.4). These text-level diagnostics do not certify valid derivations or distinct reasoning strategies.

Analysis. **Preserved coverage, efficient sampling and stable generation are separate properties.** Expert pruning can retain coverage without the length inflation seen under severe FFN removal, yet still need many more attempts. Conversely, a model can terminate more often without recovering successful answers. A useful pruning range must therefore preserve reasoning coverage while keeping the work and behavior of generation acceptable.

5 Pruned-Large versus Native-Small Models

The preceding results compare each child with its own parent. When substantial pruning preserves coverage, a second reference becomes relevant. Can the child retain broader reasoning than a native smaller model, even after losing some of its source's performance? Prior work compared large sparse and small dense models using single-sample accuracy (Jaiswal et al., 2024). We examine reasoning coverage under repeated sampling, adding the unpruned source where available to distinguish retained advantage from improvement due to pruning.

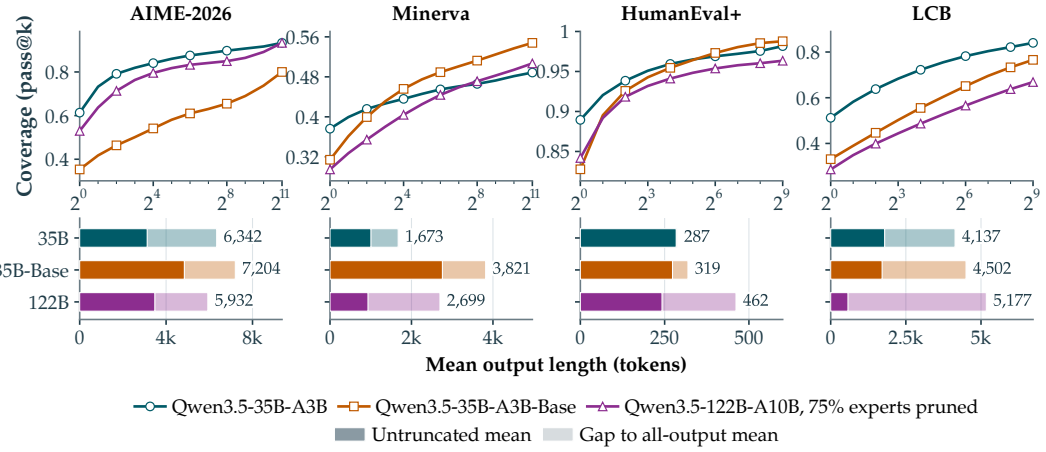


Figure 4: **Pruned-large versus native-small models.** Qwen3.5-122B-A10B with 75% expert removal versus native Qwen3.5-35B-A3B and its Base release. Parameters and training compute are not matched. The top row shows pass@k at $n = 2048$ for mathematics and 512 for code, with a separate coverage range per column, and the bottom row shows token means read as in Figure 2 (Appendix A.5).

Obs. 6. Pruning a larger model can retain broader reasoning coverage than a native smaller model, but the retained advantage depends on the source, pruning ratio and task. Figure 4 compares Qwen3.5-122B-A10B after 75% expert removal with native Qwen3.5-35B-A3B releases. The child matches Instruct on AIME at $k = 2048$ (93.3%) and exceeds it on Minerva (50.7% versus

48.9%). It trails on HumanEval+ and LiveCodeBench at $k = 512$ (96.3% versus 98.2% and 66.9% versus 84.0%), and native Base has the highest Minerva endpoint (54.8%). Thus, the mathematics comparison with Instruct is favorable without establishing a cross-task or cross-release advantage.

The source model explains what advantage is being retained. The dense controls compare all three models. With 5% FFN removal, Qwen3.5-9B reaches 90.0% on AIME at $K = 2048$ and 75.4% on LiveCodeBench at $K = 256$, below its own parent's 96.7% and 77.1% but above native 2B's 73.3% and 38.9%. A child can therefore lose some of its parent's coverage and still be the stronger reasoning model in both domains. This light-pruning comparison does not match FFN width (Appendix B.5.3).

More severe pruning tests how far that advantage survives. At 25% FFN removal, 9B matches native 4B's intermediate width. Its Minerva coverage rises from 54.3% to 56.6%, slightly above 4B's 54.8%, although its pass@1 remains lower (26.4% versus 33.8%). On AIME, the same child's coverage falls from 96.7% to 60.0%, below 4B's 90.0%. At 50% FFN removal, the width-matched child also trails native 2B on both mathematics endpoints (Figure 10). A larger starting model supplies an advantage to preserve, not a guarantee that every removal rate preserves it.

These comparisons answer different parts of the source-advantage question. The dense controls show what pruning retains relative to the parent, whereas the 122B comparison lacks that reference and only ranks the resulting child against native releases. The common 8,192-token cap also leaves open whether longer individual traces would change the ranking. More candidates do not resolve that question, which requires matched length-cap sweeps. Neither FFN-width matching nor nominal expert removal matches total parameters or training compute.

Analysis. **A pruned model need not improve on its parent to outperform a native smaller model.** The light-pruning controls demonstrate this opportunity in mathematics and code. What matters is whether the source's advantage exceeds the reasoning lost during compression. The width-matched and MoE comparisons show where that advantage disappears or depends on the native release. This is evidence about reasoning coverage, not equal-resource superiority.

6 Discussion

6.1 Are the Changes Consistent with Uniform Degradation?

The results so far show that pruning changes more than the frequency of successful outputs. We now test a simple explanation for the observed coverage losses. Would reducing success uniformly across the parent's problems produce the same outcome? We independently retain each recorded parent success with a common probability chosen to match the child's mean accuracy. The resulting coverage prediction is not fitted to the child's higher-budget performance (Appendix D.1).

Obs. 7. A mean-accuracy-matched uniform-loss reference does not fully account for the observed coverage losses across mathematics and code. On AIME at $K = 2048$, Qwen3.6-35B-A3B FFN-40% attains 63.3% coverage against the reference's 92.1%, a residual of -28.7 percentage points with a paired-problem 95% interval of $[-43.7, -14.1]$. The same child also falls below the reference on HumanEval+ (-8.7 points at $K = 512$) and LiveCodeBench (-46.0 points at $K = 256$, with no interval reported for this unequal-pool comparison). Gemma-4-26B-A4B-it expert-50% provides another LiveCodeBench example (-45.7 points, $[-51.9, -38.7]$).

Analysis. **The amount of accuracy lost does not fully describe how coverage changes.** The reference and child have matched mean accuracy but different coverage, so a common reduction in success frequency misses part of the pruning effect. The negative residuals in mathematics and code motivate examining which problems remain accessible, rather than only averaging their success rates (Appendix C.1). This rejects the specified uniform-loss explanation without identifying a unique alternative mechanism.

6.2 What Should We Rethink About Pruning?

Preservation has more than one meaning. A child may retain coverage while succeeding less often, or finish more outputs without solving more problems. Similar coverage can also conceal

retained, lost and newly solved cases (Appendix C.1). These distinctions explain why no single curve or endpoint summarizes the effect of pruning. The core expert advantage is a useful starting point, but its rate and selection criterion must still be assessed across the intended tasks.

Accessible reasoning is not the same as reliable use. Pass@k asks whether a successful candidate is available, not whether it can be selected without the benchmark grader. Accepted answers likewise do not certify valid derivations (Lightman et al., 2024). Sampling work and generation stability add practical information, but do not measure deployed latency or hardware use. Appendices B.8, E and E.1 develop these distinctions. Together, they locate what the present experiments establish about reasoning after pruning and what is still needed to use that reasoning reliably.

7 Related Work

Structural compression. Width methods use coupled pruning, compensation or dimensional reduction (Ma et al., 2023; An et al., 2024; van der Ouderaa et al., 2024; Ashkboos et al., 2024). Depth methods remove or collapse blocks (Men et al., 2025; Gromov et al., 2025; Song et al., 2024; Yang et al., 2024), and training-based pruning can jointly target multiple structures (Xia et al., 2024). Routed-model work prunes (Lu et al., 2024; Lasby et al., 2026) or merges experts (He et al., 2023; Jha et al., 2026), while FFE compares selection, iterative removal and recovery (Jaiswal et al., 2025).

Pruning under test-time scaling. Layer-pruning studies measure sequential budgets and pass@k up to 32, with SFT recovery and generation analyses (Wang et al., 2026). Unstructured pruning can preserve TTS performance depending on sparsity allocation (Monjur et al., 2026), and ShortOPD links block pruning to sampling recovery and suffix repetition before applying on-policy distillation (Zhang et al., 2026). We instead compare training-free structural axes, backbones and tasks using larger candidate pools without recovery training.

Coverage, allocation and selection. Repeated sampling separates success frequency from sampled solution range (Chen et al., 2021; Brown et al., 2024; Yue et al., 2025), and allocation methods govern how inference is spent (Snell et al., 2025; Muennighoff et al., 2025). Chain-of-thought prompting (Wei et al., 2022), self-consistency (Wang et al., 2023) and verifiers (Lightman et al., 2024) concern elicitation or selection rather than oracle availability. Capability suites (Jaiswal et al., 2024), output changes (Dutta et al., 2024) and joint performance–reliability–efficiency evaluation (von Rad et al., 2026) motivate looking beyond aggregate accuracy and perplexity.

8 Conclusion

Test-time scaling reveals what reasoning remains accessible after pruning and how readily it can be reached. Coverage can survive or exceed the original model's even when success becomes less frequent. Expert removal preserves reasoning most consistently in the core grid, while its selection criterion and removal rate shape both coverage and sampling efficiency. Retained coverage can require much greater token work, and stable termination alone does not imply successful reasoning. Where coverage falls, a uniform decline in success does not fully explain the loss. Pruned-large models can also retain broader reasoning than native smaller models, although the advantage varies with the source and task. Together, these findings distinguish making a model smaller from making its reasoning uniformly weaker or cheaper.

AI use statement

We used generative AI tools for language polishing, literature search, coding assistance and refinement of the presentation and analysis of recorded results. The authors are responsible for the final text, claims and artifacts.

Ethics statement

This study evaluates existing language models on mathematics, reasoning and code benchmarks. Pruning does not remove the risks of unreliable outputs. Favorable aggregate scores can obscure degradation on particular tasks or problems, especially in small evaluations such as the 30-problem AIME set and the four-candidate knowledge pilot.

The measurements also have limits that matter for their use. Accepted-answer coverage does not certify valid reasoning. Generated tokens do not measure latency, memory or hardware efficiency, and offline allocation does not establish deployment guarantees. Nominal removal rates do not match parameters or compute, while pruned-large and native-small comparisons match at most FFN width. The results therefore do not certify that a pruned model is reliable or economical to serve. We report coverage, sampling efficiency and stability together and state the scope of each comparison to make these distinctions explicit.

Reproducibility statement

Appendix A records the model releases, pruning operators and calibration, the FFN scorer, the evaluated cohorts and task subsets, the decoders, output caps and pool sizes, and the grading conventions behind every number in the main text, along with the unavailable historical versions and the untested corrected-scoring checkpoints. Appendix D gives the resampling procedure and the sampling assumptions behind every reported interval. Appendix B.9 and Appendix B.5.3 document the sampling-work and reference-model conventions used in the corresponding main-text claims. Appendix E records what remains unvalidated, including reasoning annotations, valid-strategy comparisons, full decoder sweeps, independent-seed replication and code tests that were not independently rerun.

References

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Fluctuation-based adaptive structured pruning for large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(10):10865–10873, 2024. doi: 10.1609/aaai.v38i10.28960. URL https://ojs.aaai.org/index.php/AAAI/article/view/28960.

Saleh Ashkboos, Maximilian L. Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. SliceGPT: Compress large language models by deleting rows and columns. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/316648eb8b4ffb6010f531b07848c300-Abstract-Conference.html.

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. doi: 10.48550/arXiv.2407.21787. URL https://arxiv.org/abs/2407.21787.

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. doi: 10.48550/arXiv.2107.03374. URL https://arxiv.org/abs/2107.03374.

Abhinav Dutta, Sanjeev Krishnan, Nipun Kwatra, and Ramachandran Ramjee. Accuracy is not all you need. In *Advances in Neural Information Processing Systems*, volume 37, pp. 124347–124390, 2024. doi: 10.52202/079017-3950. URL

https://proceedings.neurips.cc/paper_files/paper/2024/hash/e0e956681b04ac126679e8c7dd706b2e-Abstract-Conference.html. arXiv:2407.09141.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Daniel A. Roberts. The unreasonable ineffectiveness of the deeper layers. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/cbabc2f70de2dd09f491a8715ec3e80f-Abstract-Conference.html.

Shwai He, Run-Ze Fan, Liang Ding, Li Shen, Tianyi Zhou, and Dacheng Tao. Merging experts into one: Improving computational efficiency of mixture of experts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14685–14691, 2023. doi: 10.18653/v1/2023.emnlp-main.907. URL https://aclanthology.org/2023.emnlp-main.907/.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://iclr.cc/virtual_2020/poster_rygGQyrFvH.html.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida I. Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/94074dd5a072d28ff75a76dabed43767-Abstract-Conference.html.

Ajay Jaiswal, Zhe Gan, Xianzhi Du, Bowen Zhang, Zhangyang Wang, and Yinfei Yang. Compressing LLMs: The truth is rarely pure and never simple. In *International Conference on Learning Representations*, 2024. doi: 10.48550/arXiv.2310.01382. URL https://iclr.cc/virtual/2024/poster/19229. arXiv:2310.01382.

Ajay Jaiswal, Jianyu Wang, Yixiao Li, Pingzhi Li, Tianlong Chen, Zhangyang Wang, Chong Wang, Ruoming Pang, and Xianzhi Du. Finding fantastic experts in MoEs: A unified study for expert dropping strategies and observations. *arXiv preprint arXiv:2504.05586*, 2025. doi: 10.48550/arXiv.2504.05586. URL https://arxiv.org/abs/2504.05586.

Saurav Jha, Maryam Hashemzadeh, Ali Saheb Pasand, Ali Parviz, Min-Joong Lee, and Boris Knyazev. REAM: Merging improves pruning of experts in LLMs. *arXiv preprint arXiv:2604.04356*, 2026. doi: 10.48550/arXiv.2604.04356. URL https://arxiv.org/abs/2604.04356.

Mike Lasby, Ivan Lazarevich, Nish Sinnadurai, Sean Lie, Yani Ioannou, and Vithursan Thangarasa. REAP the experts: Why pruning prevails for one-shot MoE compression. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/ed93b2b5722acc2341d421b8916404a1-Abstract-Conference.html. arXiv:2510.13999.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 3843–3857, 2022. doi: 10.52202/068431-0278. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/18abbeef8cfe9203fdf9053c9c4fe191-Abstract-Conference.html.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri 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.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21558–21572, 2023. doi: 10.52202/075280-0943. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/43e9d647ccd3e4b7b5baab53f0368686-Abstract-Conference.html.

Xudong Lu, Qi Liu, Yuhui Xu, Aojun Zhou, Siyuan Huang, Bo Zhang, Junchi Yan, and Hongsheng Li. Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6159–6172, 2024. doi: 10.18653/v1/2024.acl-long.334. URL https://aclanthology.org/2024.acl-long.334/.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. LLM-Pruner: On the structural pruning of large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0950. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/44956951349095f74492a5471128a7e0-Abstract-Conference.html.

Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. ShortGPT: Layers in large language models are more redundant than you expect. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20192–20204, 2025. doi: 10.18653/v1/2025.findings-acl.1035. URL https://aclanthology.org/2025.findings-acl.1035/.

Ocean Monjur, Shahriar Kabir Nahin, and Anshuman Chhabra. Revisiting the effectiveness of LLM pruning for test-time scaling. *arXiv preprint arXiv:2604.25098*, 2026. doi: 10.48550/arXiv.2604.25098. URL https://arxiv.org/abs/2604.25098v3. Accepted to Findings of EMNLP 2026; version 3.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1025. URL https://aclanthology.org/2025.emnlp-main.1025/.

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.

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.

Jiwon Song, Kyungseok Oh, Taesu Kim, Hyungjun Kim, Yulhwa Kim, and Jae-Joon Kim. SLEB: Streamlining LLMs through redundancy verification and elimination of transformer blocks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 46136–46155. PMLR, 2024. URL https://proceedings.mlr.press/v235/song24f.html.

Mingyang Song and Mao Zheng. RAZOR: Pruning replaceable experts in LLMs, 2026. URL https://github.com/nick7nlp/Razor. Preprint.

Tycho F. A. van der Ouderaa, Markus Nagel, Mart van Baalen, Yuki M. Asano, and Tijmen Blankevoort. The LLM surgeon. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/38a1671ab0747b6ffe4d1c6ef117a3a9-Abstract-Conference.html.

Jonathan von Rad, Yong Cao, and Andreas Geiger. UniComp: A unified evaluation of large language model compression via pruning, quantization, and distillation. *arXiv preprint arXiv:2602.09130*, 2026. doi: 10.48550/arXiv.2602.09130. URL https://arxiv.org/abs/2602.09130v6. Accepted to EMNLP 2026; version 6, revised August 28, 2026.

Keyu Wang, Tian Lyu, Guinan Su, Lu Yin, Marco Canini, Jonas Geiping, and Shiwei Liu. When fewer layers break more chains: Layer pruning harms test-time scaling in LLMs. In *Conference on Language Modeling*, 2026. doi: 10.48550/arXiv.2510.22228. URL https://openreview.net/forum?id=eAi51zoI7n. arXiv:2510.22228, version 2.

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. arXiv:2203.11171.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pp. 95266–95290, 2024. doi: 10.52202/079017-3018. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ad236edc564f3e3156e1b2feafb99a24-Abstract-Datasets_and_Benchmarks_Track.html.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024. doi: 10.1093/jrsssb/qkad009. URL https://academic.oup.com/jrsssb/article/86/1/1/7043257.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. Sheared LLaMA: Accelerating language model pre-training via structured pruning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/160adf2dc118a920e7858484b92a37d8-Abstract-Conference.html.

Yifei Yang, Zouying Cao, and Hai Zhao. LaCo: Large language model pruning via layer collapse. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 6401–6417. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.372. URL https://aclanthology.org/2024.findings-emnlp.372/.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1933. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/537d5aa768c2d534016a4d06f87bc8fb-Abstract-Conference.html.

Qingyu Zhang, Qianhao Yuan, Hongyu Lin, Yaojie Lu, Xianpei Han, Le Sun, Ming Xu, and Jiarui Li. ShortOPD: Recovering pruned LLMs with short-to-long on-policy distillation. *arXiv preprint arXiv:2607.13124*, 2026. doi: 10.48550/arXiv.2607.13124. URL https://arxiv.org/abs/2607.13124v2. Version 2, revised August 14, 2026.

APPENDIX GUIDE

The main text organizes the evidence as local observations and scoped analyses. The appendices expose the supporting material in five roles. These are experimental specification, complete model-level evidence, problem and trajectory analyses, statistical support and robustness, and extended interpretation. This separation keeps the narrative focused while preserving controls, negative results and measurement limits. The statistical tools support the observations rather than constituting separate contributions.

Layer	Purpose and contents	Read
Experimental details	Architectures, pruning operators and calibration, evaluated cohorts, task subsets, sampling, grading and measurement conventions.	A
Extended model-level results	Complete MoE and dense sweeps, expert selection, native-small comparisons, cross-task retention, stability, target attainment and cumulative token use.	B
Problem and trajectory analyses	Retained, lost and newly solved problems. Success-frequency changes, derivation validity, trace length, repetition and surface diversity.	C
Statistical support and robustness	Uniform-loss references, difficulty-order tests, uncertainty, aggregate matching, larger pools, decoding and run sensitivity, proofs and sampling assumptions.	D
Extended discussion and open questions	Implications for pruning, measurement limits, the knowledge pilot, secondary allocation diagnostics and remaining validation needs.	E

Evidence scope. Problem transitions, resampling, statistical references and offline allocation reuse recorded candidate pools and add no independent generations. Appendix analyses refine the main observations within those pools. Surface-text measurements still do not certify valid reasoning or replace the annotations and replication discussed in Appendix E.

APPENDIX CONTENTS

A Experimental Details and Measurement **15**

A.1 Experimental roles and measured coverage 15

A.2 Model architectures and parameter scales 17

A.3 Pruning operators, removal fractions and calibration 18

A.4 Sampling, answer grading, and code execution 20

A.5 Metrics and comparison conventions 20

A.6 Evaluation sources and limitations 21

B Extended Model-Level Results **22**

B.1 Common-budget MoE profiles 22

B.2 Broader primary collection and endpoint summaries 23

B.3 Four-task sweeps of the Gemma evaluations 24

B.4 Dense-backbone coverage, output length and stability 24

B.5 Backbone, selection and native-small comparisons 27

B.6 Cross-task coverage and common-budget decisions 30
B.7 Generation stability across pruning axes 31
B.8 Available candidates, target attainment and answer extraction 32
B.9 From output length to cumulative token use and resource limits 34

C Problem- and Trajectory-Level Analyses **35**
C.1 Retained, lost and newly observed problems across pruning rates 35
C.2 Success-frequency transitions 35
C.3 A correct answer with an invalid derivation 36
C.4 Token efficiency within and across trajectories 37
C.5 Diversity of valid solution strategies 40

D Statistical Support and Robustness **41**
D.1 Can uniform success loss explain the coverage changes? 41
D.2 Does pruning preserve the ordering of problem difficulty? 43
D.3 Paired-problem uncertainty 44
D.4 Extending Minerva sampling beyond 2,048 candidates 45
D.5 Sensitivity to decoding and trajectory-length limits 45
D.6 Same-seed run sensitivity 47
D.7 Technical support for finite-pool references and sensitivity 48
D.8 Technical support for ordering certificates and sampling assumptions 49
D.9 What information do the curves identify? 51

E Extended Discussion and Open Questions **54**
E.1 What the joint profile implies for pruning choices 54
E.2 What does accepted-answer coverage establish? 54
E.3 A bounded knowledge-task pilot 55
E.4 Can changed success frequencies inform candidate allocation? 56
E.5 Limits of independent-seed replication 59
E.6 Remaining validation needs 59

A Experimental Details and Measurement

This appendix documents experimental roles and coverage, model architectures, pruning operations, sampling and grading, and comparison conventions. It distinguishes observed candidate coverage from single-sample retention, target attainment and answer selection, and records the limitations of the evaluation sources. Paired-problem uncertainty and statistical sensitivity checks are collected in Appendix D.

A.1 Experimental roles and measured coverage

Core MoE sweeps and Gemma follow-up. Table 1 covers three unpruned MoE parents and 57 pruned checkpoints. The 48 Qwen and GLM children form the core three-axis sweeps, while nine

Gemma expert children extend the backbone comparison. The common analysis pools contain 2,048 candidates per problem for each checkpoint on both AIME-2026 (30 problems) and Minerva. The Minerva pool covers the 219 retained problems for the Qwen and GLM children and all 272 for the Gemma children, whose grading was corrected as described in Appendix A.4. A checkpoint denotes one recorded weight configuration, not one task, report, sampling budget or decoding intervention. All available primary RAZOR, Taylor FFN and perplexity-greedy depth checkpoints for each displayed model are included. The AIME pools for Qwen expert-25%, 30% and 40% were adopted after inspecting completed runs. Comparisons involving these pools are post-selection descriptions, not independent confirmatory replications. Within this reference cohort, Gemma contributes only an expert sweep. Its additional FFN and depth evaluations in Appendix B.3 are reported separately and do not enter these cohort counts. The aggregate analyses use an earlier, explicitly specified subset, as detailed below.

Table 1: **Core and follow-up MoE coverage with measured removal labels.** Rates are nominal percentages, not verified parameter or FLOPs reductions. Every listed child is paired with its own unpruned parent at $n = 2048$ on both mathematics tasks. Parent rows are not included in the child counts.

Model / pruning axis	Nominal removal (%)	Children
Qwen3.6-35B-A3B		
Expert / RAZOR	10, 20, 25, 30, 40, 50, 60, 70, 75	9
FFN / Taylor	5, 10, 15, 20, 25, 30, 40, 50	8
Depth / PPL	10, 20, 30, 40, 50, 60	6
GLM-4.7-Flash		
Expert / RAZOR	10, 20, 25, 30, 40, 50, 60, 70, 75	9
FFN / Taylor	5, 10, 15, 20, 25, 30, 40, 50	8
Depth / PPL	4, 11, 13, 15, 19, 25, 38, 50	8
Gemma-4-26B-A4B-it		
Expert / RAZOR	10, 20, 25, 30, 40, 50, 60, 70, 75	9
FFN / Taylor	5, 10, 15, 20, 25, 30, 40, 50 (outside cohort)	—
Depth / PPL	20, 40 (outside cohort)	—
Total	7 measured model–axis cells	57

Gemma's FFN and depth rows are measured on the recorded protocol at $n = 2048$ and are drawn in Figures 5–6 and 8, but they are follow-ups rather than reference-cohort members and are excluded from the 57-child total and from every aggregate computed over it. Depth pruning removes atomic six-layer units from thirty layers. Only 20% and 40% removal were evaluated, with 60% not pursued after degenerate output at 40%.

Dense and selection controls. Qwen3.5-9B, Qwen3.5-4B, Qwen3.5-2B and Llama-3.1-8B-Instruct supply supplementary FFN and depth comparisons. These are separate dense parent models, not dense-converted versions of the three MoEs. Their checkpoints are paired with their own parents. Tail-drop provides a depth-selection control, not an additional point in the main PPL sweep. Expert-activation norm (EAN) and routing frequency provide four additional Qwen expert-selection comparisons. They are not pooled with RAZOR in the main figures.

Budget eligibility and missing evaluations. Six primary tail-drop runs were screened at $n = 256$ and were not extended by the original evaluation. They support $\mathrm{pass@}k$ only for $k \leq 256$. Their unmeasured $\mathrm{pass@2048}$ remains missing. The broader full-budget collection is consequently selected by that screening procedure, not an unbiased sample of possible pruned models. Mathematics and code also cover different weight configurations. A retention ratio requires a same-backbone parent on that task and a matched supported budget, and each checkpoint is read at its own available budget. Repetition-penalty sweeps reuse weights and are not counted as additional pruning checkpoints.

Analysis denominators. The reference mathematics collection holds 127 pruned children, 121 of them with $n = 2048$ mathematics pools and six at $n = 256$. They partition into 57 MoE checkpoints

(48 from the core sweep and nine Gemma follow-ups), 42 dense Taylor and PPL checkpoints, 24 tail-drop controls on MoE and dense parents, and four additional expert criteria. These counts describe weight configurations, not task runs. The seven unpruned parents are counted separately, and the Gemma FFN and depth curves of Appendix B.3 fall outside them. The aggregate analyses use the earlier 125-child subset, which excludes the subsequently added Gemma expert-60% and expert-70% evaluations. This subset has 119 full-budget children. Excluding its four additional expert-criterion rows gives 117 full-budget primary children. The two added rates appear in the extended sweeps, but not in the reference fits, uncertainty calculations, ordering certificates or allocation summaries. From that primary set, excluding the 3 outcome-selected rows gives 114 inference-eligible children, whereas excluding the 19 Gemma children gives the 108-child full-budget primary set used in several summaries. These are different filters of the same primary collection, not successive exclusions. The 100-child common-budget cohort further requires paired evidence on all three tasks and complete mathematics pools (Appendix B.6.2). These denominators identify eligibility sets, not additional experiments or independent replications.

A.2 Model architectures and parameter scales

Table 2 summarizes the seven unpruned parent models, focusing on parameter scales and the layer, FFN and expert structure relevant to the pruning comparisons. The four Qwen releases are post-trained models despite the absence of an `-Instruct` suffix in their official names. The suffix `-it` in Gemma-4-26B-A4B-it identifies the instruction-tuned variant, as does `-Instruct` in Llama-3.1-8B-Instruct. Architectural dimensions refer to the language decoder, excluding vision encoders and auxiliary multi-token-prediction (MTP) modules. "Dense" and "MoE" describe the feed-forward network (FFN), not the attention layout.

Table 2: **Parent-model scales and pruning-relevant structure.** Parameter scales are official nominal values in billions (B), not exact tensor counts. d_{model} is the residual-stream width, and d_{FFN} is the dense-FFN width or, for MoE models, one routed expert's width. Shared paths are specified separately in (b).

(a) Nominal parameter scales and decoder dimensions

Model	FFN type	Total (B)	Active (B)	Layers	d_{model}	d_{FFN}
Qwen3.6-35B-A3B	MoE	35	3	40	2048	512
GLM-4.7-Flash	MoE	30	3	47	2048	1536
Gemma-4-26B-A4B-it	MoE	26	4	30	2816	704
Qwen3.5-9B	Dense	9	—	32	4096	12288
Qwen3.5-4B	Dense	4	—	32	2560	9216
Qwen3.5-2B	Dense	2	—	24	2048	6144
Llama-3.1-8B-Instruct	Dense	8	—	32	4096	14336

(b) Expert routing and shared FFN paths

Model	MoE layers	Routed experts/layer	Routed top-k	Shared paths/layer	Shared width
Qwen3.6-35B-A3B	40	256	8	1	512
GLM-4.7-Flash	46	64	4	1	1536
Gemma-4-26B-A4B-it	30	128	8	1	2112

Reading the table. A dash means that a separate active-parameter scale is not reported for the dense model. GLM-4.7-Flash has one initial dense FFN of width 10240 followed by 46 MoE layers. Its listed d_{FFN} is the routed-expert width. Routed top-k excludes the shared path and is distinct from decoding top-k.

Parameter conventions. Gemma-4-26B-A4B-it uses the official 26B/A4B designation, with detailed model-card entries of 25.2B total and 3.8B active. Inclusion of its separately listed vision encoder is unspecified, so no encoder adjustment is made. The four Qwen cards report language-

model parameters. An expert-removal percentage is not the same percentage reduction in total parameters, active parameters or FLOPs.

Specification sources and model identity. The released model cards are Qwen/Qwen3.6-35B-A3B, zai-org/GLM-4.7-Flash, Gemma-4-26B-A4B-it (Google), Qwen/Qwen3.5-9B, Qwen/Qwen3.5-4B, Qwen/Qwen3.5-2B, and meta-llama/Llama-3.1-8B-Instruct (accessed September 12, 2026). Architecture values were checked against local configurations for the first six models. Llama-3.1-8B-Instruct architecture values use a public Llama-3.1-8B-Instruct configuration mirror from Unsloth, because the original evaluation configuration is unavailable. The Gemma-4-26B-A4B-it configuration comes from a later copy of the same release. These sources establish architectural specifications, but do not fully verify the weights used in historical evaluations.

A.3 Pruning operators, removal fractions and calibration

What a removal percentage means. We distinguish the nominal requested fraction $r \in [0, 1]$, reported as $100r\%$, from the realized structural removal fraction. Expert pruning removes routed experts within each MoE layer, giving $r_{\text{expert}} = 1 - E_{\text{kept}}/E_{\text{parent}}$. The denominator excludes shared experts and dense FFNs, which this axis does not remove. FFN pruning removes intermediate channels inside each dense, routed-expert or shared FFN, giving a per-module fraction $r_{\text{FFN},u} = 1 - d_{u,\text{kept}}/d_{u,\text{parent}}$. It leaves the residual-stream width d_{model} unchanged. Depth pruning removes trunk layers, $r_{\text{depth}} = 1 - L_{\text{kept}}/L_{\text{parent}}$, with the original total trunk depth as denominator, not only the eligible layers. Auxiliary prediction heads are accounted for separately. Thus 25% on different axes does not mean equal total-parameter, active-parameter or FLOPs savings.

All sweep plots use nominal removal labels. Realized fractions are available only for the Gemma expert checkpoints. The RAZOR labels 10, 20, 25, 30, 40, 50 and 75% retain respectively 115, 102, 96, 90, 77, 64 and 32 of 128 routed experts per layer, so the 10% label is $13/128 = 10.15625\%$ removal and the 30% and 40% labels are 29.6875% and 39.84375%. The recorded kept-index sets fix these counts, and the 90 and 77 kept experts round up from 89.6 and 76.8 rather than truncating to 89 or 76. The remaining checkpoints lack verified realized fractions. In particular, rounding GLM's nominal depth labels would not establish the actual layer counts used during evaluation.

Axis labels. Expert, FFN and depth denote the three structural axes. "Layer" in the coverage and length grids denotes depth pruning, not a separate intervention. Where selection criteria are shown, RAZOR, Taylor and PPL identify the primary expert, FFN and perplexity-greedy depth methods, respectively.

Expert selection. RAZOR is the primary expert-selection criterion used in this study (Song & Zheng, 2026). It scores the local output change when a routed expert is removed and the next-ranked expert fills its slot. For token t, let $\mathcal{K}_t$ be the parent's routed top-k support and ρ_t its next-ranked expert under the model's routing rule. Let g_{tj} denote mixing weights before support normalization, $Z_t = \max(\sum_{j\in\mathcal{K}_t} g_{tj}, 10^{-20})$, and $u_{tj} = g_{tj}/Z_t$, including $j = \rho_t$. The replacement weight therefore uses the original support's normalizer. With $y_t = \sum_{j\in\mathcal{K}_t} u_{tj} f_j(x_t)$, the score is

$$d_{ti} = \frac{\lambda \left\| u_{ti}(f_i(x_t) - y_t) - u_{t\rho_t}(f_{\rho_t}(x_t) - y_t) \right\|_2}{\max(1 - u_{ti} + u_{t\rho_t}, 10^{-6})}, \qquad s_i = \left[\frac{\sum_{t\in\mathcal{T}_i} d_{ti}^2}{\max(|\mathcal{T}_i|, 1)} \right]^{1/2}. \tag{3}$$

Here $\mathcal{T}_i$ contains valid calibration tokens routed to expert i, λ is the model's routed-output scaling factor, and model-specific expert gains are included in f_i. The RMS average is conditional on routing to the expert. Experts with the smallest scores are removed within each MoE layer. This is a single-expert replacement surrogate at fixed layer input, not the end-to-end effect of jointly removing multiple experts. The available RAZOR procedure preserves the parent's routed top-k unless an alternative target is specified, but the historical routing settings are not fully recoverable. Routed top-k is unrelated to decoding top-k.

The expert-activation norm (EAN) control averages $\|f_i(x_t)\|_2$ over the same expert's routed tokens, while frequency selection uses the valid routed-token count. These are related to activation-norm and routing-frequency baselines in recent expert-pruning comparisons (Lasby et al., 2026), but the

aggregations here are the specific exported implementations being evaluated, not every possible implementation of those criteria.

The available expert-calibration protocol uses one fixed calibration corpus, held constant across every checkpoint, with model-specific chat templates, shuffling at seed zero and packs of up to 32,768 tokens. In this packing mode the corpus is consumed once, and the number of packs depends on the tokenizer. The pack length is a calibration setting, not the generation cap. The calibration corpus is not publicly available, which limits exact replication of expert selection. Kept-index records support the retained-set comparison even where the original saliency values are unavailable. Reconstructed ordering bands are not used as substitutes for the missing saliency magnitudes.

FFN selection and width alignment. The FFN criterion selects intermediate channels using the sum of parameter-wise Taylor saliencies over the gate, up, and down projections,

$$I_j = \sum_h |W_{jh}^{\text{gate}} G_{jh}^{\text{gate}}| + \sum_h |W_{jh}^{\text{up}} G_{jh}^{\text{up}}| + \sum_h |W_{hj}^{\text{down}} G_{hj}^{\text{down}}|, \tag{4}$$

where G denotes the gradients used for scoring. Channels are selected independently inside each dense FFN, routed expert and shared FFN. For the widths in this study, the available procedure keeps $d_{u,\text{kept}} = 8\,\text{round}[d_{u,\text{parent}}(1-r)/8]$, subject to valid-width bounds. This alignment can yield different realized fractions for modules with different original widths. The available calibration settings request 16 or 32 chunks of 512 calibration tokens, but cannot be reconstructed uniformly for every historical checkpoint. The available scorer clears gradients within each chunk iteration and reads them after the loop, retaining only the final chunk's gradient. Historical scorer versions are not available for every checkpoint. The results therefore characterize the evaluated Taylor-ranked checkpoints, not pruning with verified full-corpus gradient accumulation or a complete reproduction of LLM-Pruner.

Depth selection and atomic units. Following SLEB's iterative loss-based block selection (Song et al., 2024), perplexity-greedy depth pruning recomputes the loss after each removal. At each step it bypasses each remaining admissible unit together with the units already removed, and selects the candidate giving the smallest teacher-forced calibration perplexity. A unit is the minimal complete attention-pattern cycle, or a single layer in a homogeneous architecture. Index-pinned leading dense layers are protected. The available calibration procedure prepares up to 64 chunks of 2,048 calibration tokens and evaluates 16 chunks per candidate. Selected units are then physically removed, preserving the relative order of the surviving layers. Tail-drop instead removes late admissible units while retaining the final unit. A nominal 25% removal need not be representable under these atomic units.

Calibration amounts and scoring procedures differ between axes, and some historical settings are unavailable. Matching nominal removal or single-sample accuracy loss therefore does not provide a controlled comparison of equally optimized pruning algorithms.

Calibration-gradient sensitivity. The last-chunk scoring issue motivates a comparison with the mean signed gradient over the same calibration chunks. For chunk losses $\ell_1, \ldots, \ell_B$, the two criteria use

$$G_{\text{last}} = \nabla_W \ell_B, \qquad G_{\text{mean}} = \frac{1}{B} \sum_{b=1}^{B} \nabla_W \ell_b.$$

Under the corrected criterion, the signed gradients are averaged before the absolute value in Eq. 4 is applied. This differs from averaging absolute per-chunk saliencies and permits cancellation across chunks.

A toy example shows that this distinction can change the channel ranking. With unit weights and chunk gradients $(-10, 1)$ and $(2, 3)$ in each of three projections, last-chunk saliencies are $(6, 9)$, whereas mean-gradient saliencies are $(12, 6)$. The selected channel is reversed. This algebraic example shows sensitivity of the criterion, not a change in LLM accuracy. No checkpoint has yet been rescored and evaluated under the corrected criterion. The example also cannot determine which unavailable historical scorer version produced a particular checkpoint.

A.4 SAMPLING, ANSWER GRADING, AND CODE EXECUTION

Mathematics uses temperature 0.7, top-p 0.95, decoding top-k 20, a maximum of 8,192 generated tokens, and chunks of 32 samples. The base random seed is 42. Fixed chunk seeds make prescribed smaller budgets nested within larger ones when the same run is extended. A candidate prefix is defined by chunk indices, not completion order. Mathematics and code use different seed formulas. Separate runs using the same seeds are not independent-seed replications.

AIME uses all 30 problems. Minerva begins with 272 reference problems, of which 53 have reference answers that the mathematical grader cannot parse. Metrics on the retained set use the same 219 problem IDs for each checkpoint, with one exception noted below.[1] The grader processes the completion text rather than treating failure to terminate as an automatic incorrect score. A truncated sample can already contain a gradeable answer, and a terminated sample can contain none.

For mathematical majority voting, answers are normalized into buckets. An unparseable answer contributes no vote, and ties are resolved randomly. Smaller budgets are estimated by 1,000 resamples without replacement from the available candidate pool. These are estimates of the implemented normalized-answer selector, not a general claim about semantic equivalence among derivations. The oracle and selector both inherit the limitations of the answer parser.

HumanEval+ (Liu et al., 2023) uses 164 problems and the base-plus test suite. LiveCodeBench (Jain et al., 2025) uses the available local 175-problem subset and executed tests. This subset does not establish a contamination-free evaluation. Their principal budgets are 512 and 256 samples per problem. Code majority-vote accuracy is unavailable and is not inferred from textual agreement among generated programs.

A.5 METRICS AND COMPARISON CONVENTIONS

We use $P_m(k)$ for the finite-pool pass@k statistic in Eq. 1 and 0 for the unpruned parent of the same backbone. The available pool size is n (or n_q when it varies by problem), the selected budget is k, and K is the endpoint budget of the comparison. In Eq. 2, $E_m = P_m(1)/P_0(1)$ measures single-sample accuracy retention and $R_{m,K} = P_m(K)/P_0(K)$ measures same-budget coverage retention. A zero parent denominator makes the ratio undefined. In particular, E_m is not single-trajectory chain-of-thought (CoT) efficiency.

Units and missing values. Scores P and majority accuracy are displayed as percentages, while E and R are dimensionless parent-normalized ratios, displayed either as ratios on figure axes or as percentages in tables. A difference between percentage scores is in percentage points (pp), not percent relative change. Differences are taken before rounding, so subtracting two displayed scores can miss the reported difference by one unit in the last shown digit. Counts of checkpoints or problems are not sample-pool sizes. A dash denotes an undefined or unavailable quantity where specified, and a numerical zero is a measured value. Reference-specific N/A labels retain the reason the comparison cannot be formed.

Checkpoint labels. In compact diagnostic tables, Qwen (or Qwen3.6), GLM and Gemma denote Qwen3.6-35B-A3B, GLM-4.7-Flash and Gemma-4-26B-A4B-it, respectively. The 9B, 4B and 2B labels denote the corresponding Qwen3.5 models, not parameter-matched versions of a pruned model. Table 2 gives the full model specifications. A rate following Expert, FFN or Depth is a nominal removal percentage unless explicitly labeled actual.

Two supplementary comparisons use different references. These are fixed-parent target attainment and majority-vote extraction. Same-k retention compares the child and parent at the same selected budget, whereas a fixed parent target uses the parent's endpoint throughout.

To express compensation in units of sampling, set a common target $y_\alpha = \alpha P_0(K)$ and define

$$k^*_m(\alpha) = \min\{k \in \{1, \ldots, K\} : P_m(k) \geq y_\alpha\}. \tag{5}$$

[1]The Gemma checkpoints reported in this paper were identified as affected by a parser exclusion that incorrectly dropped otherwise-gradeable answers, and were rescored under a corrected grader that admits all 272 Minerva problems. Gemma numbers throughout this paper reflect that 272-problem grading. Other backbones' Minerva numbers still use the original 219-problem retained set pending the same correction.

We use $\alpha = 0.9$, with 0.75 and 1.0 as sensitivity checks. Unattained targets are right-censored at K. When both curves attain y, define the empirical multiplier $S_m(y) = k_m(y)/k_0(y)$ using their smallest integer budgets. Integer rounding alone can vary this ratio, so small-budget fluctuations need not indicate shape distortion.

Majority-vote extraction uses mathematical majority voting under the answer normalization and tie rules in Appendix A.4. It is not defined for the code results, for which no majority-vote evaluation is available.

For a checkpoint with $P_m(K) > P_m(1)$, define

$$X_{m,K} = \frac{\text{maj@}K_m - P_m(1)}{P_m(K) - P_m(1)}. \tag{6}$$

The denominator is the same model's oracle headroom. X measures the fraction converted into accuracy by majority voting and may be negative. A zero headroom gives an undefined ratio, not zero extraction. We report the underlying three scores and compare X with the same parent rather than pooling an unmatched collection of parent ratios.

Supplementary model-origin comparison and its length statistic. Figure 4 uses a separate Qwen3.5 comparison, not the core Qwen3.6 grid. It uses the native 35B-A3B and 35B-A3B-Base releases and a 122B checkpoint with 75% expert removal. All three reports specify the decoding settings listed in Appendix A.4, together with repetition penalty 1.0 and a no-thinking template request. Mathematics uses 2,048 candidates per problem and code 512. This comparison and the additional expert-selection controls do not enlarge the primary-cohort denominators. Metadata agreement neither proves identical template behavior nor verifies historically served weights.

The lower row reports all-output and conditional untruncated means for each model and task. For the mean length L_k of k candidates, a stationary sampling policy gives $\mathbb{E}[L_k] = \mathbb{E}[L]$, so expected mean length is constant in k rather than cumulative token use.

The untruncated mean is reconstructed from the all-output mean $\bar{t}$ and truncation fraction r as $(\bar{t} - 8192r)/(1 - r)$, assuming capped outputs contain exactly 8,192 tokens. It describes a selected subset, not an uncapped counterfactual. The pale segment is the difference between the all-output and untruncated means, not the capped outputs' additive contribution to the all-output mean or their unobserved continuation length.

On LCB, pruned 122B has the longest all-output mean but the shortest untruncated mean. Its truncation rate is 60.4%, versus 36.6% and 43.0% for native Instruct and Base. Thus the untruncated subset comprises 39.6% of the pruned model's outputs, versus 63.4% for Instruct, so the conditional means compare differently selected outputs.

A.6 Evaluation sources and limitations

Analyses use recorded scores, fixed parent pairings and explicit eligibility criteria. Mathematics identities, success counts and sample counts are checked against reported aggregates, while code metrics are recomputed from per-problem counts, without independently rerunning tests. Code-verification claims exclude evaluations without executed tests, and comparisons requiring complete coverage exclude incomplete records. Missing evaluations are not measured zero scores.

We use corrected reruns where the inference service had served the wrong checkpoint, but cannot establish the weights served for every historical request. Gemma's sampled retained-expert tensors agree with the evaluated parent only over the probed layers, tensor types and experts. An alternative historical set of weights fails those probes and is not used as the parent.

Gemma uses a newer language-model serving stack than the original backbones. The requested no-thinking template setting does not ensure identical template behavior across models. Some early mathematics reports also lack a recorded repetition penalty. These values remain unknown, and a protocol-sensitivity analysis separates the affected reports from those with complete settings. The comparisons therefore emphasize within-backbone changes rather than isolated architecture effects. Timing records affected by score reaggregation are not used to infer throughput or GPU cost.

Gemma RAZOR at 50% has complete score records, finish reasons and token counts, but incomplete saved text. It remains eligible for score-based comparisons. Analyses requiring complete trajectories exclude it rather than extrapolating from the available text.

B Extended Model-Level Results

These results extend the main-text comparisons with complete pruning curves, selection and model-origin controls, cross-task results and generation stability. Fixed-parent targets, majority-vote extraction and cumulative token use distinguish candidate availability from answer selection and resource proxies. Cohort definitions follow Appendix A.1, while matching and extended-sampling checks are collected in Appendix D.

B.1 Common-budget MoE profiles

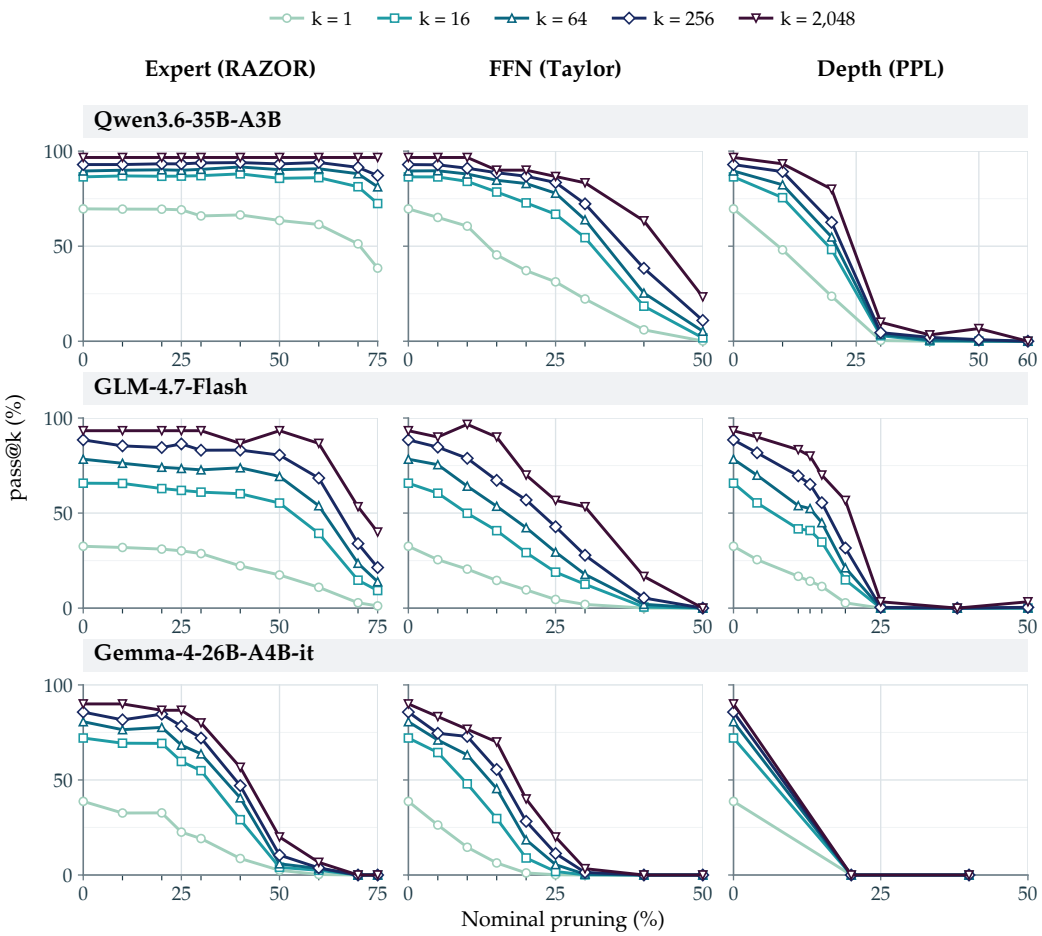


Figure 5: **How each sampling budget responds to pruning on AIME-2026.** Each line fixes $k \in \{1, 16, 64, 256, 2048\}$ and traverses all measured nominal rates within one model–axis cell. The unpruned parent is at zero removal. Absolute pass@k uses the same 0–100% scale across panels. The profiles include Gemma expert, FFN and depth pruning and end at each cell's largest measured rate. Nominal rates do not match computational savings. The profiles are discussed in Section 3.2, with Gemma's grid specified in Appendix A.1.

Figure 1 presents the 48-child core mathematics sweeps with the two expert-25% children omitted from display only. Figures 5–6 retain all 67 core and follow-up children in the severity slices, including Gemma's nine expert, eight FFN and two depth checkpoints. Gemma's FFN and depth screens are follow-ups outside the reference cohort and do not enter the 48-child core count or any

aggregate derived from it. Appendix B.2 covers the broader primary collection, including dense backbones, and endpoint summaries.

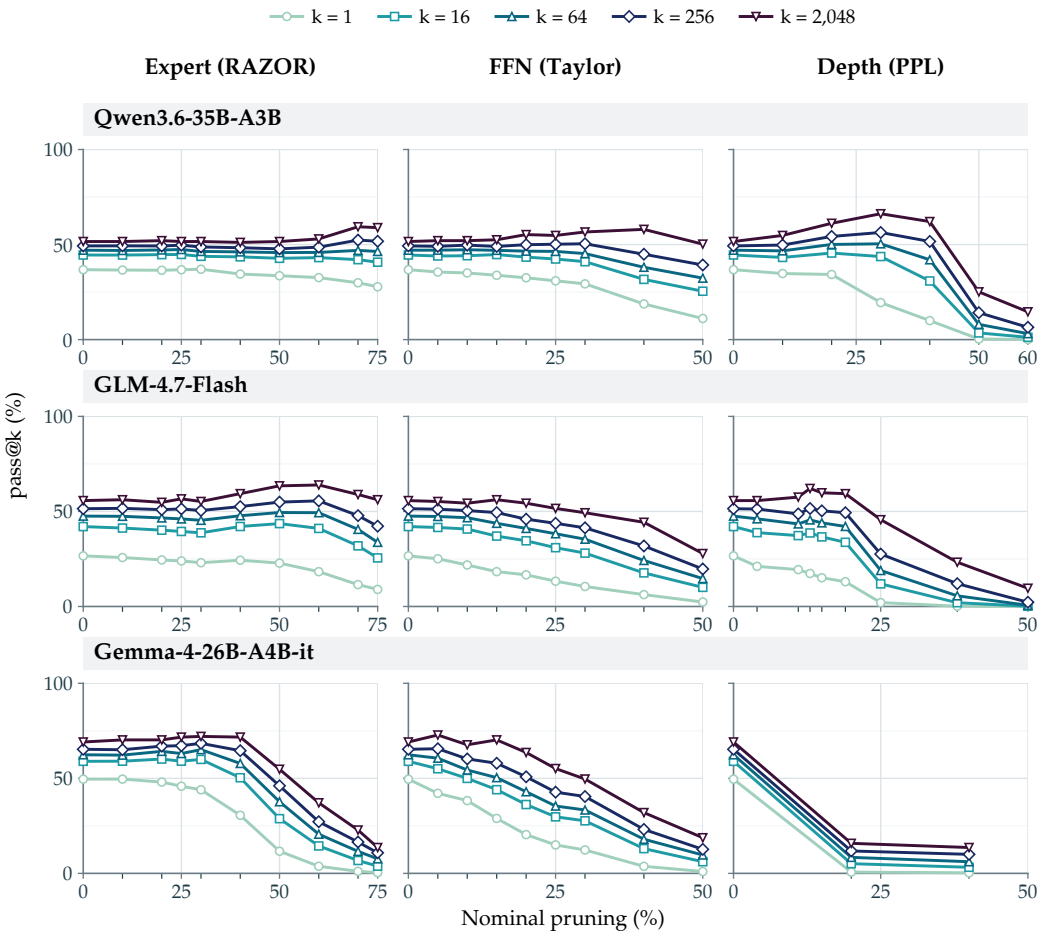


Figure 6: **Minerva severity slices at five sampling budgets.** The layout and budgets match Figure 5. Every plotted rate is measured, and the unequal spacing of GLM depth labels is preserved. Comparing these profiles with AIME separates task-dependent coverage changes from a claim of a universal pruning threshold.

B.2 Broader primary collection and endpoint summaries

Figure 7 plots the observed endpoints. Distance above its diagonal is a normalized retention contrast, not an amount of capacity or a guarantee of recovery. The cohort is the full-budget primary collection excluding Gemma, over unequal sets of backbones and rates. Expert pruning is available only for MoEs, whereas width and depth include dense models, so a difference between axes pooled this way is not an architecture-controlled effect size. The same-backbone comparisons in the main text address a narrower question.

AIME endpoints at $n = 2048$ are counts out of 30, so one additional observed problem changes absolute coverage by 3.33 percentage points. Relative changes also depend on how many problems the parent covers. Matching the parent's count does not establish that the same problems were covered, and a one-problem improvement is not taken as evidence that pruning created a new ability. The per-problem transitions in Appendix C retain this distinction.



Figure 7: **Single-sample and large-budget retention.** E and R_{2048} use the same parent reference. The diagonal marks equal relative retention, not equal absolute scores or inference cost.

B.3 Four-task sweeps of the Gemma evaluations

Figure 8 extends the visual comparison in Figure 1 to the additional Gemma evaluations. Columns show AIME-2026, Minerva, HumanEval+ and LiveCodeBench, with 30, 272, 164 and 175 evaluated problems, respectively. The Minerva column uses the corrected 272-problem grading of Appendix A.4. Each row names a pruning axis and selection criterion at its right end. Children share the evaluated problem set with their task-specific Original. Colors, markers and line styles identify nominal removal rates, following the dense-backbone layout in Appendix B.4.

The first two rows compare the primary expert criterion with REAP. Their matched-budget coverage differences are reported in Section 3.3. REAP was screened at $n = 256$ samples per problem, whereas the primary mathematics curves use $n = 2048$, so comparisons between them stop at the common $k = 256$. The REAP Minerva pools are mixed, because the 219 problems the original parser admitted carry 256 samples and the 53 the corrected grader added carry 2,048. Each problem's coverage is estimated from its own pool, and the $k = 256$ ceiling is set by the shallower 219.

The grid is complete in this render, because the primary expert sweep contributes nine removal rates on each of the four tasks, REAP three, FFN eight and depth two. Each curve stops at its observed pool size, so the two code columns end at $k = 256$. No panel is empty, and a curve held at zero is therefore a measured result without successes rather than an unavailable evaluation. The depth children are the clearest case, because neither the 20% nor the 40% layer-pruned child solves any HumanEval+ or LiveCodeBench problem in 256 samples, and neither solves any AIME problem in 2,048, yet both still reach 15.8% and 13.6% Minerva coverage at $k = 2048$. Repeated sampling therefore recovers nothing on three of the four tasks for this axis, which is a stronger statement than any single task would support. These expanded displays do not change the reference-cohort aggregates defined in Appendix A.1.

B.4 Dense-backbone coverage, output length and stability

Figures 9–12 extend the coverage and output-length comparisons to Qwen3.5-2B, Qwen3.5-4B, Qwen3.5-9B and Llama-3.1-8B-Instruct. Each backbone contributes FFN and layer sweeps across the same four tasks, and routed-expert pruning does not apply to these dense models. Parent references remain specific to the backbone and task. As in Figure 1, the ordinate of every coverage panel is the same fraction of evaluated problems and is named once per pair of rows rather than once per row.

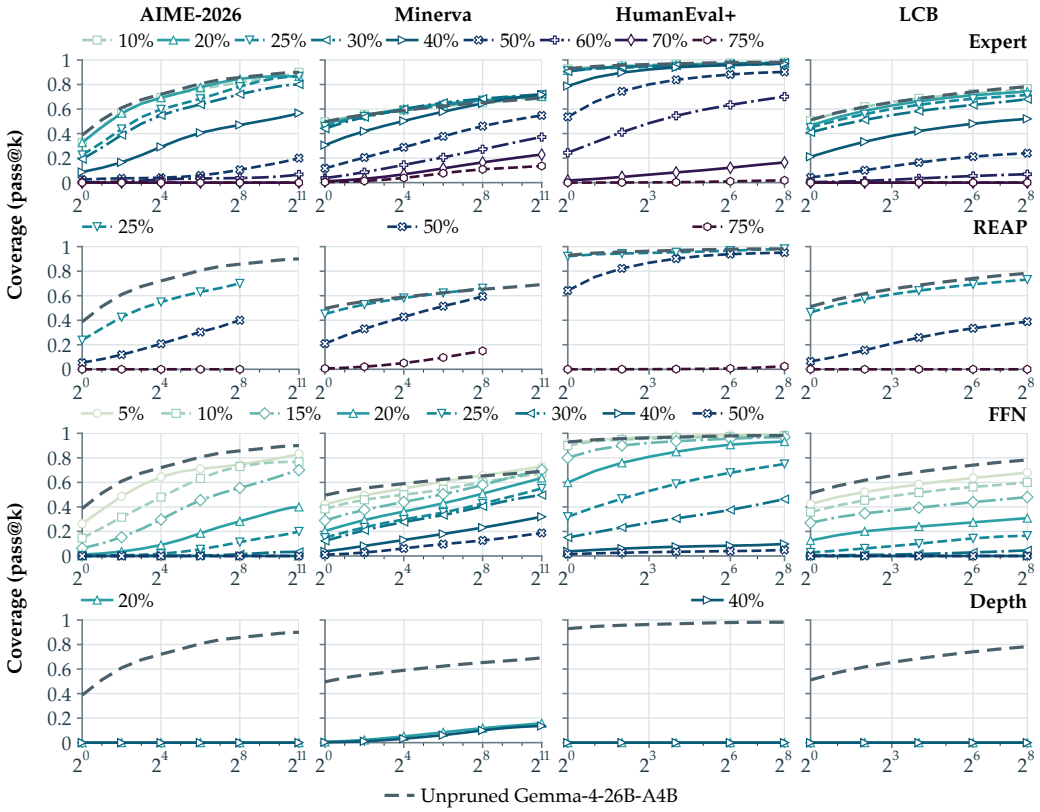


Figure 8: **Gemma-4-26B-A4B-it across four reasoning tasks and three pruning axes.** Rows show expert pruning under the primary criterion, expert pruning under REAP, FFN and depth. Each child is compared with its same-task Original. Minerva uses the corrected 272-problem grading (Appendix A.4), while AIME-2026, HumanEval+ and LiveCodeBench cover 30, 164 and 175 problems. REAP curves stop at $k = 256$, while the other mathematics curves continue to $k = 2048$. Code curves stop at their observed pools. Every panel is measured, so a curve held at zero records samples without successes rather than a missing evaluation. Coverage uses the ordinate convention of Figure 1. Removal percentages are nominal structural rates, not matched parameter or computation reductions.

Shared plotting conventions. Each backbone has FFN and layer rows, and task columns follow Figure 1. The protocol follows Section 2.2. Repetition-penalty and long-context variants of Qwen3.5-9B are excluded. Mathematics is displayed through $k = 2048$, and code curves stop at their own pools. Zero curves denote completed evaluations without successes, not missing results. Colours encode ordered rates within each figure, and the main and dense grids have sixteen and fourteen entries, respectively. Compare labelled percentages, not colours across figures. Removal is nominal rather than resource-matched.

Unequal removal grids. Qwen3.5-2B and Qwen3.5-4B use layer ladders of $17/33/50\%$ and $12/25/50\%$, respectively. Llama-3.1-8B-Instruct has eight layer rates, resolving coverage collapse more finely than the Qwen backbones' three- and four-rate ladders. A grid whose first measured rate is already beyond the collapse cannot locate its onset. Panel alignment therefore implies neither matched severity nor, for code, a common maximum sampling budget.

Output length and termination. The length grids retain the coverage grids' rows and columns and use Figure 3's encoding, with all-sample and conditional untruncated means on the left axis and pass@1 and pass@256 on the right. The dotted line marks the 8,192-token cap, and the grey reference is the parent's all-sample mean. Open grey rings mark truncation of at least 20%. Labelled ticks give measured rates, with shorter ticks between them, and each row ends at its own largest rate.

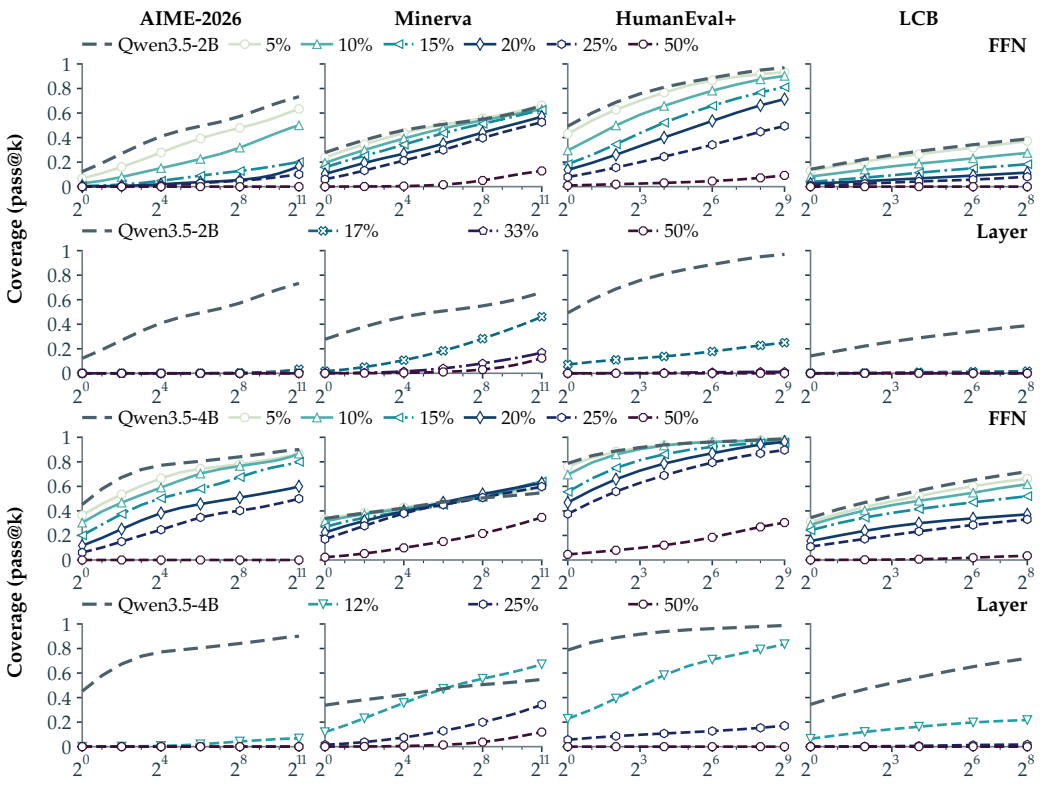


Figure 9: **Dense-model coverage for Qwen3.5-2B and Qwen3.5-4B.**

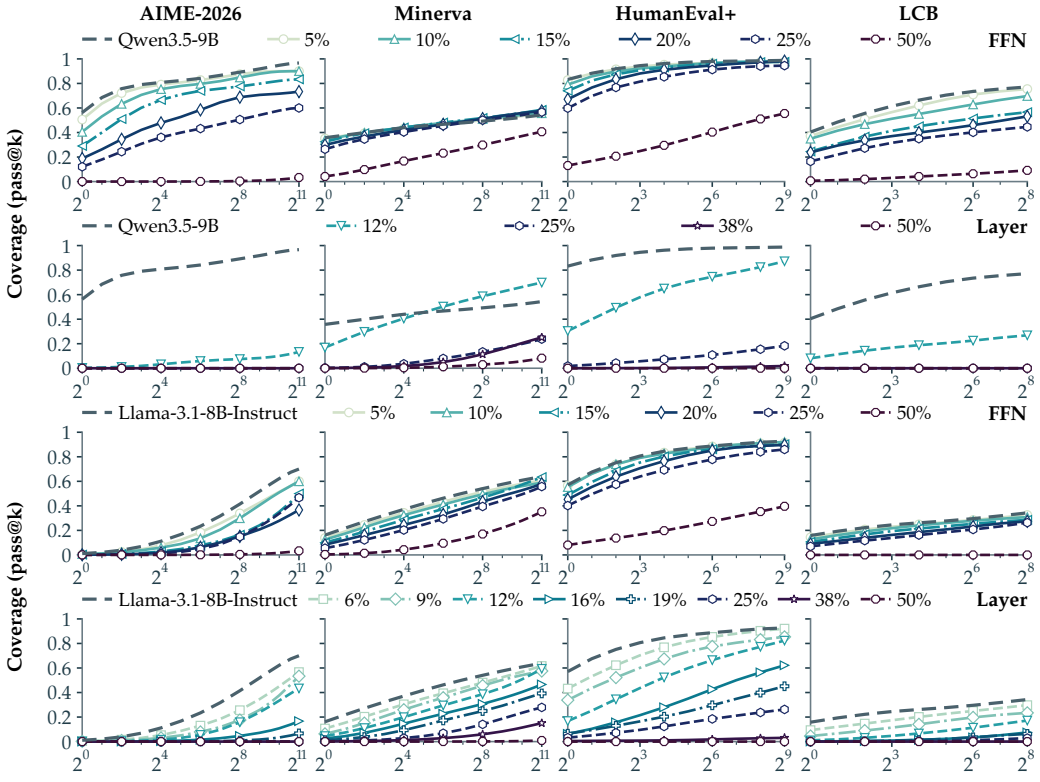


Figure 10: **Dense-model coverage for Qwen3.5-9B and Llama-3.1-8B-Instruct.**

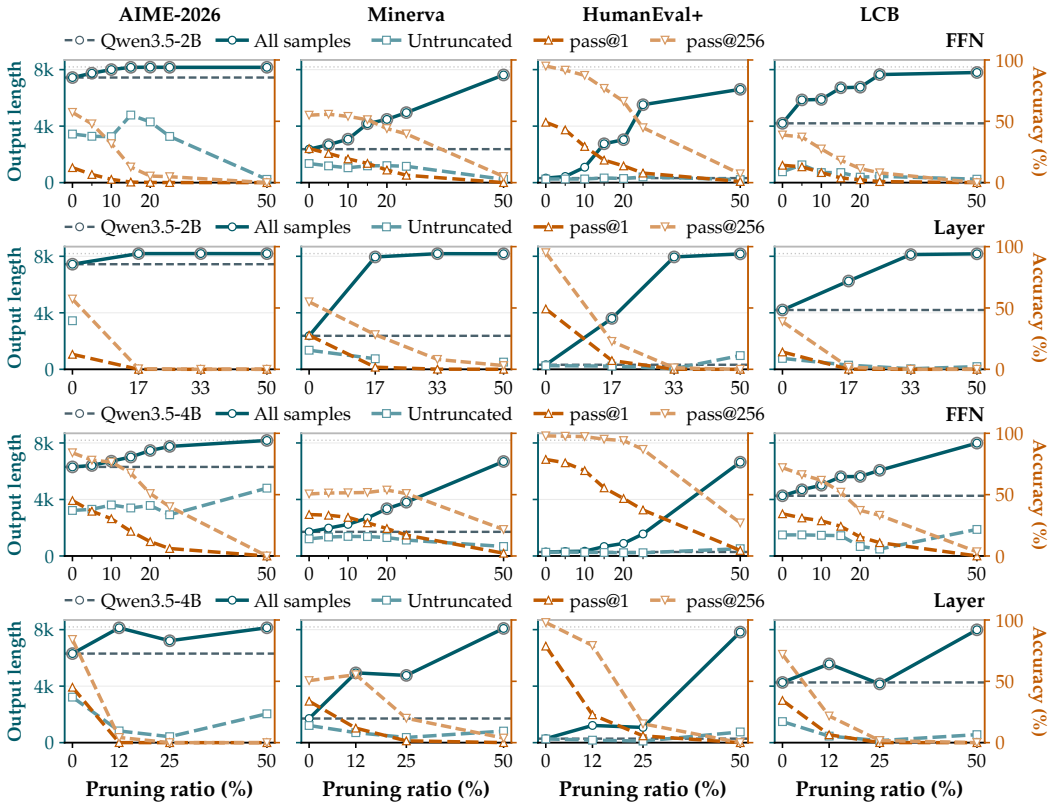


Figure 11: **Dense-model token use for Qwen3.5-2B and Qwen3.5-4B.**

Lengths are not conditioned on correctness. A rising mean alongside falling pass@1 records longer output, not necessarily more reasoning. Near the cap, the untruncated mean describes a shrinking, selected subset rather than the lengths that all outputs would have had without truncation.

B.5 Backbone, selection and native-small comparisons

These comparisons test the breadth of the observed patterns across backbones and alternative pruning selections. The dense native-small comparisons match recorded FFN widths only, while the supplementary MoE comparison contrasts model origins without a verified total-parameter match. Neither matches training compute or isolates an architecture-only causal effect.

B.5.1 Expert sweeps across backbones

Figure 13 compares the nine-rate Qwen3.6-35B-A3B sweep with the seven Gemma-4-26B-A4B-it RAZOR rates in the analysis subset. Gemma's additional 60% and 70% evaluations appear in Figures 5, 6 and 8. At 75% removal, Qwen retains its parent's same 29 AIME problems, whereas Gemma reaches none versus its parent's 27 within 2,048 samples. At 50%, Qwen still covers 29 while Gemma reaches six (20.0%). Qwen's preserved range is not a general expert-pruning property. Gemma's serving stack and high parent truncation remain relevant alternative explanations. These comparisons do not isolate architecture effects.

Gemma's expert sweep moves from partial compensation to pronounced coverage loss. At 25% removal, AIME single-sample accuracy falls from 38.7% to 22.6% while coverage changes from 90.0% to 86.7%. At 50%, the two scores are 2.5% and 20.0%. No correct AIME answer is observed at 75% removal. The corresponding cross-task endpoint comparison is given in Appendix B.6, and the task-specific sampling budgets should not be interchanged. GLM also loses coverage at its deepest expert tier. Retaining at least as many experts per layer as the router activates ensures routing feasibility, not expert quality.

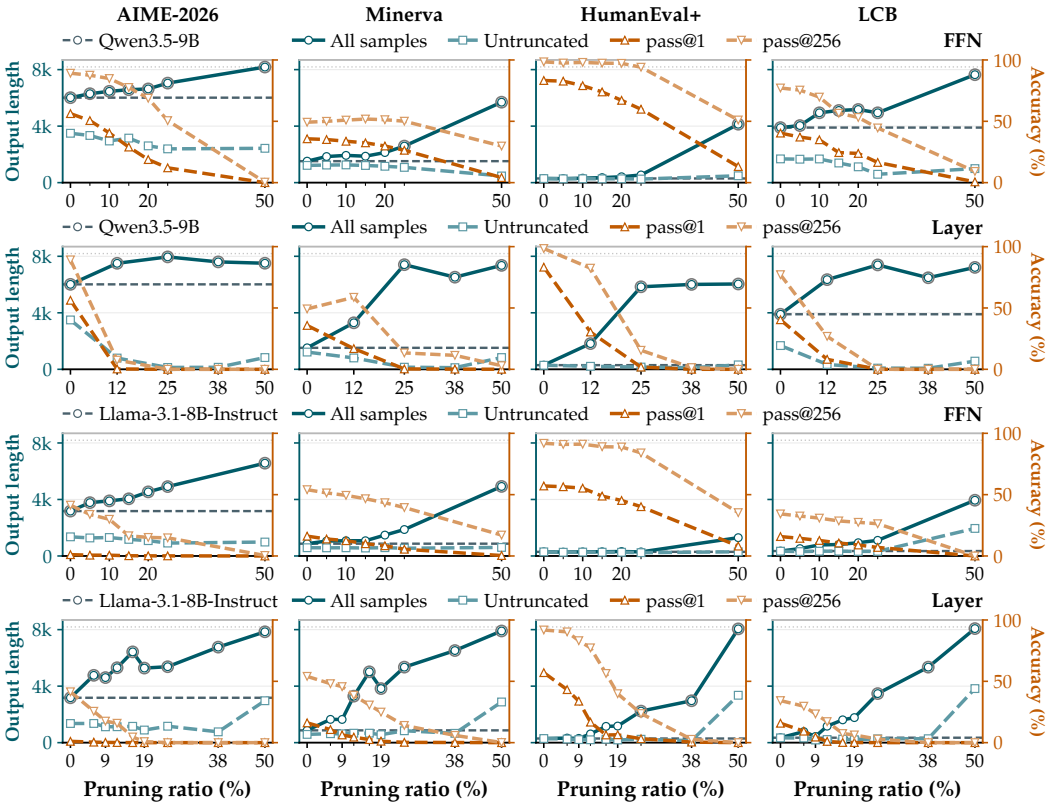


Figure 12: **Dense-model token use for Qwen3.5-9B and Llama-3.1-8B-Instruct.**

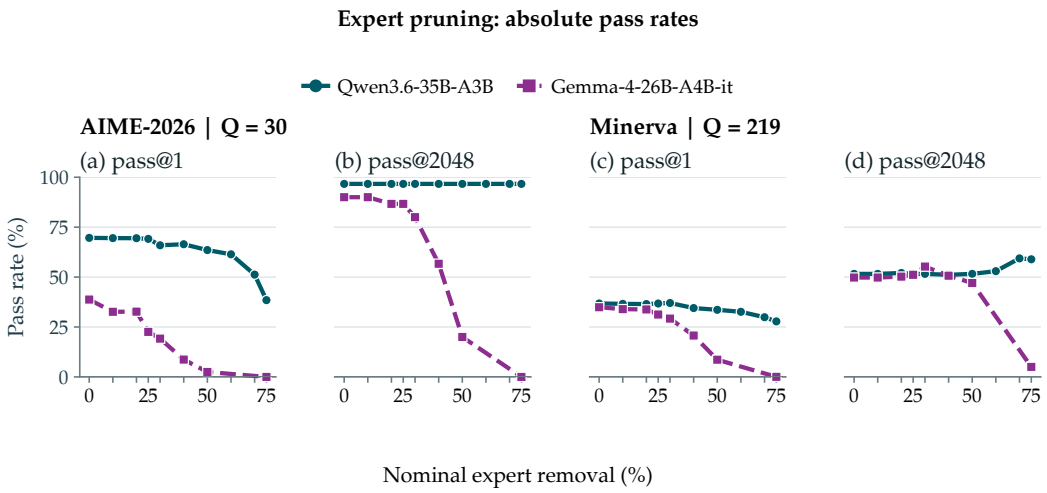


Figure 13: **The expert-pruning profile does not transfer uniformly across backbones.** Nine Qwen3.6-35B-A3B and seven Gemma-4-26B-A4B-it RAZOR removal rates are drawn, each against its own unpruned parent. The seven are the Gemma rates in the analysis subset. Its 60% and 70% screens are measured but sit outside that subset, and appear in Figure 8. Each task is shown as a pass@1 and a pass@2048 panel, both on absolute scales. All mathematics pools have 2,048 candidates per problem. Serving-stack and truncation differences preclude an architecture-only causal interpretation.

The expert counts behind Gemma's realized removal fractions are in Appendix A.3. Reconstructed score-order bands do not recover the original saliency magnitudes or the ranking inside a band.

At the common $K = 64$, Gemma expert-50% retains 91.6% of its parent's HumanEval+ score, but 7.3% on AIME and 28.6% on LiveCodeBench. Qwen FFN-20% retains 92.6%, 98.9% and 77.1% on AIME, HumanEval+ and LiveCodeBench, respectively. These within-task ratios illustrate the broader task disagreement without equating task-specific maximum budgets.

B.5.2 Pruning-selection controls

Selection also matters. GLM's nominal depth-4% reaches 10.0% AIME coverage under tail-drop versus 90.0% under calibrated selection at $K = 2048$. A separate Qwen TailExpert control at 50% removal reaches 43.3% AIME coverage versus 96.7% for each calibrated criterion at $k = 2048$. It is not displayed in the expert-selection figures. Additional Qwen expert criteria (EAN and routing frequency) are compared at their available common budget. These four rows remain outside the primary one-criterion expert grid. Small differences, including failure to reject a difference at a mild rate, are not treated as proof that the criteria are equivalent. Comparisons across multiple rates and tasks are exploratory rather than independent confirmations of a prespecified ordering.

Figure 14 compares the four calibrated criteria at 25% removal, complementing the 50% results in Figure 2. At 25%, their all-output mean lengths are approximately 0.94–1.10 times their same-task parent's mean, and truncation is close to the parent's own rate, so the split inside each bar is narrow. At 50% the same ratios span 0.89–2.29 and the Minerva lengths separate sharply even where coverage remains close. These generated-token ratios neither measure hardware cost nor establish a monotonic effect of removal. Both figures use the same task-specific token scales and common comparison budgets, where LCB has $n = 512$ for the parent and RAZOR versus $n = 256$ for the other selections. Additional selection criteria do not change the primary sweep cohort.

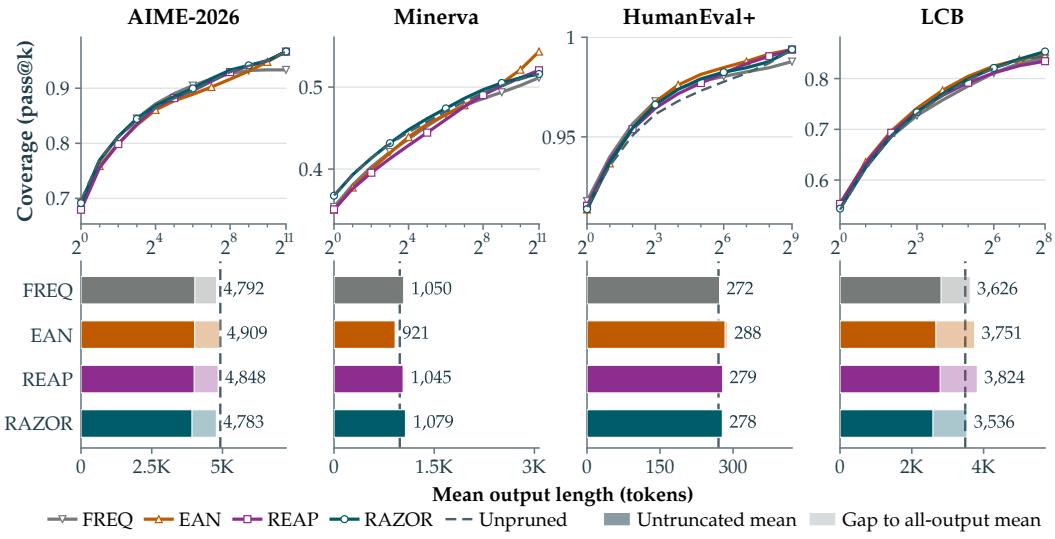


Figure 14: **Expert selection at 25% removal on Qwen3.6-35B-A3B.** The top row shows pass@k and the bottom row shows mean output length in tokens. Solid bars show conditional untruncated means, and pale extensions reach the all-output means. Methods, parent references and scales match Figure 2.

B.5.3 Native-small and pruned-large comparisons

Removing 25% or 50% of the 9B FFN width matches the recorded intermediate width of the native 4B or 2B, respectively. Total parameters, other architectural dimensions and training compute remain unmatched.

The additional 4B pruning grid is useful for within-family robustness but is not a dedicated matched-2B experiment. In particular, 50% removal from its FFN width does not reproduce the native 2B width. Both the 4B FFN-50% and depth-50% checkpoints have zero observed AIME coverage at 2,048

samples, but labeling them "4B to 2B" would make a resource-matching claim the evaluation did not establish. Lighter pruning supplies the cross-domain detail behind the retained-lead case in Section 5. With 5% FFN removal, 9B reaches 90.0% on AIME ($K = 2048$) and 75.4% on LCB ($K = 256$), below its own parent's 96.7% and 77.1% but above native 2B's 73.3% and 38.9%, respectively. This child remains substantially larger than 2B, and unlike the width-matched comparisons even FFN width is not matched, so the example establishes a retained coverage advantage rather than an explanation of its origin or resource efficiency.

The supplementary MoE comparison in Figure 4 uses the evaluated Qwen3.5-122B-A10B expert-75% checkpoint and the native Qwen3.5-35B-A3B and Qwen3.5-35B-A3B-Base releases. It is distinct from the core Qwen3.6-35B-A3B sweep. The unpruned 122B source is not available in this comparison, so the results cannot quantify how much of its coverage the child retains. The four-task results in Section 5 show that endpoint agreement on AIME does not extend uniformly to Minerva or code, and that the native release matters. These are model-origin comparisons, not controlled training-from-scratch experiments at equal parameters, activated computation or pretraining budget.

B.6 Cross-task coverage and common-budget decisions

The executed-code cohorts test cross-task coverage at their supported task-specific endpoints. A separate common-budget intersection compares task decisions. The bounded knowledge pilot in Appendix E.3 has its own limited coverage and does not extend the full mathematics or code cohorts.

B.6.1 Executed-code cohorts and task-specific endpoints

The task-specific reference cohorts contain 107 HumanEval+ and 118 LiveCodeBench children. The additional Qwen3.5-4B HumanEval+ and Qwen3.5-2B depth-33% LiveCodeBench curves in Appendix B.4 are not included in these aggregates or the reference common-budget intersection. Their endpoint budgets K differ, whereas the common-budget comparison in Appendix B.6.2 fixes the same k across tasks. Its 100 children form a different eligibility set requiring complete mathematics pools and paired evidence on all three tasks, not the union or sum of the two code cohorts.

Code provides a different verification process from numerical answer matching. Figure 15 compares the endpoint retentions of the available parent-paired cohorts. On Gemma, RAZOR at nominal 30% has 90.7% HumanEval+ single-sample accuracy, compared with 92.9% for its parent. At 512 samples their coverage is 97.6% and 98.2%, a one-problem difference. On LiveCodeBench, the corresponding single-sample scores are 41.0% and 51.1%, and coverage at 256 samples is 68.0% and 78.3%. These outcomes differ from both the mild and severe AIME regimes. A benchmark label alone does not identify the amount or type of damage.

Generation length is one potential contributor, but cross-task length differences also accompany changes in difficulty, prompts, and grading. Shorter outputs provide fewer opportunities to hit the same token cap. That observation does not show that output length measures sequential reasoning depth or causes the cross-task performance difference. Where problem-level length analyses are used, the reference is the parent's length and difficulty, rather than a pruned model's possibly degenerate output length.

B.6.2 Common-budget task decisions

The decision rule used here takes the parent at the same k, unlike the sampling target in Eq. 5, which fixes the parent at the maximum budget. At $k = 64$ and required retention 90%, the (AIME, HumanEval+, LiveCodeBench) pass patterns have counts $(111) : 19$, $(011) : 8$, $(110) : 4$, $(010) : 18$, and $(000) : 51$. The three-task intersection within the full-mathematics-budget primary subset is held fixed across the sensitivity settings. The six $n = 256$ tail-drop screens are excluded from this cohort even though they support these values of k. A missing parent or task is also excluded rather than interpreted as a failed model. Changing the rule can change both the number of acceptable checkpoints and which tasks disagree. These comparisons demonstrate decision non-equivalence under specified rules, not that one benchmark is universally harder or a particular selection policy is optimal.

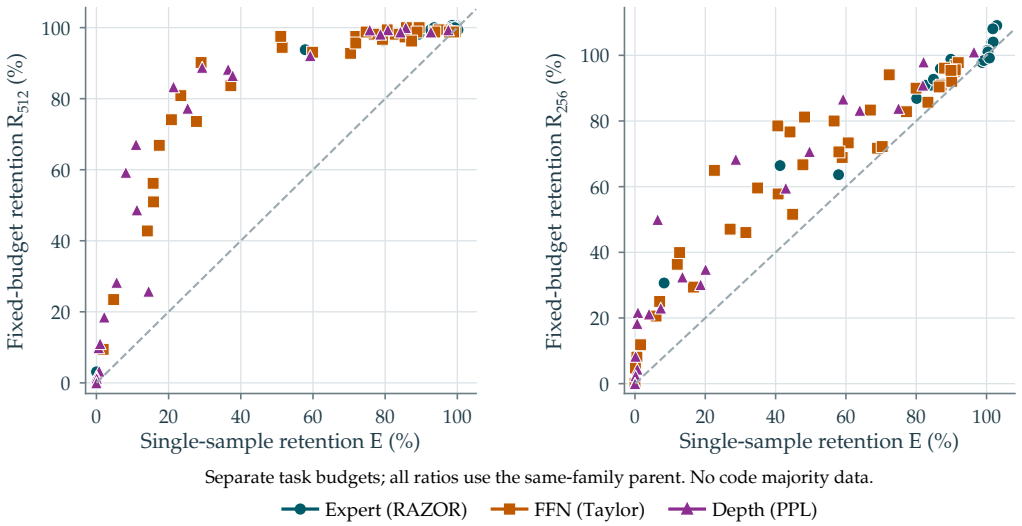


Figure 15: Primary-cohort code coverage relative to the same-task parent. HumanEval+ uses $K = 512$ and LiveCodeBench $K = 256$, with the panels shown separately. Only expert, FFN and calibrated depth pruning are displayed. Tail-drop remains in the reference cohort summaries but is omitted here. Additional selection controls, missing parents, and unsupported larger budgets are excluded. These executed-test reports test cross-task generality of the measured pruning patterns, not a universal causal relation between truncation and capability.

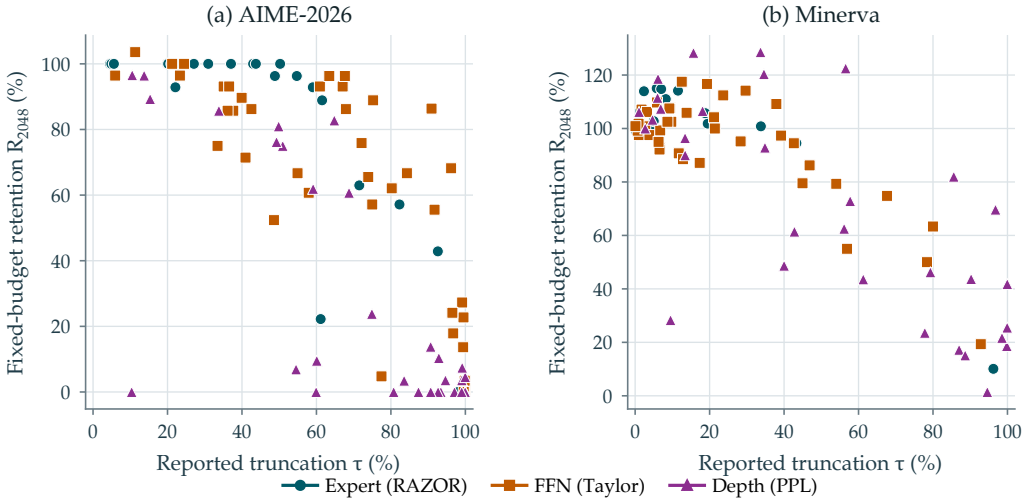


Figure 16: **Truncation and same-task coverage retention.** AIME and Minerva are shown separately for expert, FFN and calibrated depth pruning, and tail-drop is omitted from this display. Associations across checkpoints are behavioral diagnostics, not causal mediation estimates.

B.7 Generation stability across pruning axes

The global coverage comparisons are complemented by response-failure statistics. Truncation, missing answers and early repetitive termination are distinct outcomes, and their associations with accuracy do not identify a causal mediator of pruning damage or measure the validity of a reasoning trace.

Truncation rate is the fraction of samples whose finish reason is the token cap. Figure 16 shows its association with coverage retention. It can be estimated from fewer samples than a large-budget coverage endpoint, which makes it useful for screening. It is not identical to the fraction of samples without a marked answer. Some failures stop early with irrelevant text, while some cap-limited samples already contain an answer. The current grid includes Qwen3.5-9B with 50% tail-drop, which has 7.0% AIME truncation and no observed AIME success at its $n = 256$ screening budget. Low truncation is not a certificate of health.

Sample-level comparisons often find higher correctness among terminated than truncated outputs. This comparison conditions on a behavior correlated with problem difficulty, trajectory quality, and answer availability. It does not estimate the accuracy that would result from forcing the truncated samples to terminate. Similarly, a repetition-penalty intervention changes the whole token distribution, not only the stop decision. Its effect on correctness cannot be attributed to truncation alone without additional assumptions.

The available penalty sweep provides no controlled intervention that isolates termination. We therefore use the observations to describe failures, not as a two-stage causal mechanism for all pruning damage. An answer-emission screen also requires held-out validation if its threshold is selected on the same checkpoint grid. A threshold that classifies these observations well is not a calibrated deployment guarantee.

Perplexity measures a different object. Prior compression evaluation shows that small perplexity changes can coexist with downstream degradation and more repetitive free-running generations (Jaiswal et al., 2024). Our results likewise make calibration perplexity insufficient as the sole screening metric in this setting, but do not imply that perplexity contains no useful information. Likewise, empirical critical rates fitted to truncation curves describe where that behavior changes in the observed grid. A logistic extrapolation beyond the largest tested rate, or beyond the router's feasible range, cannot certify a safe compression rate.

Axis-specific response behavior. Figure 17 separates truncation, looping and cross-candidate overlap in 128-candidate-per-problem trajectory pools. On AIME, GLM expert pruning stays near its parent's 1.4% truncation through 30% removal, whereas FFN and depth pruning reach 56.6% and 99.1%, respectively, at 25% removal. Truncation falls at the deepest layer checkpoints without recovering complete-trace coverage. The terminated-error subset also changes, so its repetition statistics do not establish a transition from cap-limited generation to another failure mechanism.

The Qwen Minerva analysis uses 128 candidates per problem on 219 commonly gradeable problems. Looping is measured among correct, terminated traces and rises on the depth axis, from 1.7% to 43.9%. Cross-candidate overlap is measured among terminated traces, regardless of correctness, and falls on every axis. Both measures use saved text and whitespace tokenization. Thus degraded outputs need not converge to one repeated text. Above 50% truncation, however, the surviving traces are a strongly selected subset. These surface statistics do not establish reasoning validity, effective correction or semantic strategy diversity, and the corresponding annotations remain pending (Appendices C.4 and C.5).

B.8 Available candidates, target attainment and answer extraction

A successful candidate must first occur within the available sample budget and then be selected. These are separate conditions, because target attainment measures the first, while majority voting tests one particular answer selector.

B.8.1 Fixed parent targets

The fixed parent target and its attainment budget are defined in Appendix A.5 and Eq. 5.

The finite-pool oracle target in Eq. 5 is shared within a backbone. 90% of the Qwen parent's final AIME coverage is 87.0%. Expert-75% reaches it at 246 samples versus the parent's 20, a 12.3-fold increase. FFN-40% does not within 2,048. This is an observed-pool target budget, not an online sample-complexity guarantee, and nonattainment is censoring. The 95% paired-problem interval for the expert-75% budget is 23–824 samples. Every resample reaches the target within 2,048. FFN-40%

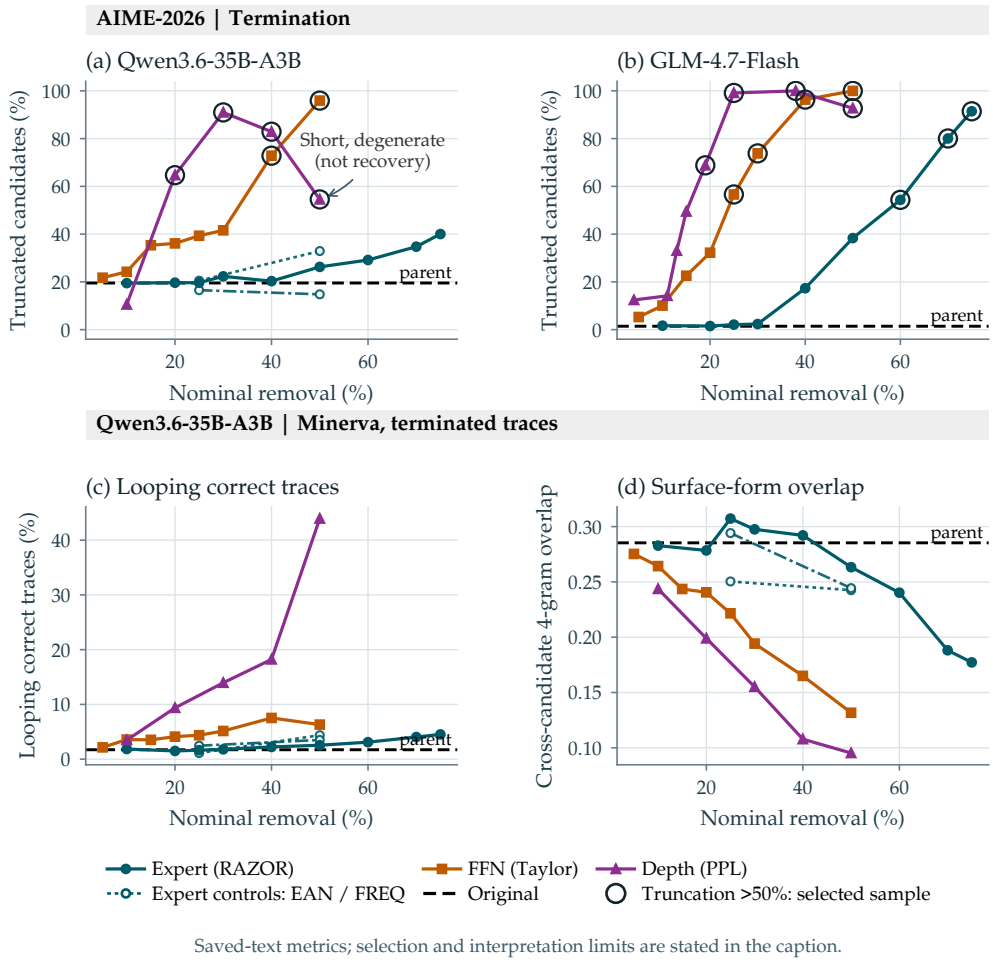


Figure 17: **Response behavior across pruning axes.** Panels (a) and (b) show AIME-2026 truncation, panel (c) shows looping among correct terminated Qwen Minerva traces, and panel (d) shows overlap among terminated traces. Colour encodes axis and line style encodes selection criterion, and alternative criteria remain separate. Circles mark truncation above 50%, where surviving traces are strongly selected. Looping and overlap use whitespace-tokenized saved text, not serving-tokenizer counts, and do not establish reasoning validity, recovery or semantic diversity.

is censored in 99.85% of resamples, and Gemma expert-50% in all resamples. These intervals condition on the recorded candidates (Appendix D.3).

Among the calibrated checkpoints of the full-budget primary cohort excluding Gemma, 16/18 expert, 11/40 FFN and 2/32 depth reach the 90% target. The depth denominator becomes 50 once the tail-drop screens are added back. The two attained checkpoints are the same in either accounting. These are descriptive counts, not severity-matched axis success rates. Censored cases are retained in the denominator at all three target fractions.

For every target fraction, the reference is the parent's score at the same final budget, not its score at the smaller attainment budget. The smallest integer attainment budget is computed from the per-problem success counts. This distinguishes a target reached late from one not reached by the evaluation cap.

The empirical $S(y)$ compares the two budgets needed for the same absolute accuracy. It is reported only on the jointly reachable target range. If the pruned curve fails to reach a target that the parent reaches, only a lower bound on its required multiplier is available. If the parent itself does not reach the target, the ratio is undefined. These cases are not assigned arbitrary large finite values for averaging or plotting.

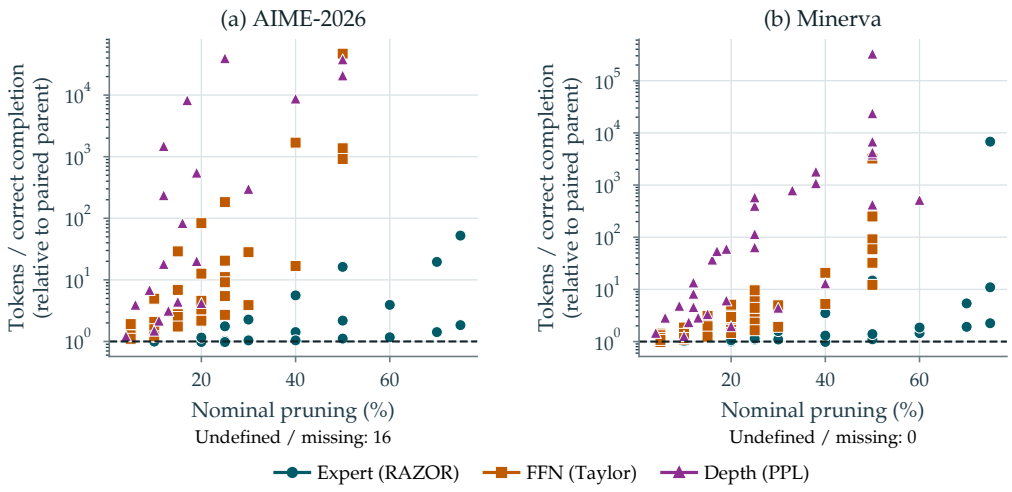


Figure 18: **Generated tokens per grader-correct completion.** Expert, FFN and calibrated depth pruning are shown, and tail-drop is omitted. Values are normalized to the same parent, and zero-success checkpoints are marked separately.

B.8.2 Oracle headroom and majority-vote extraction

The same-model oracle-headroom extraction ratio is defined in Eq. 6 (Appendix A.5).

Qwen expert pruning at 75% retains 96.7% AIME oracle coverage but majority voting reaches 66.7%, compared with 90.0% for the parent. Its extraction ratio X_{2048} falls from the parent's 75.3% to 48.5%, a 26.9-point decline. Across measured budgets, Qwen expert-75% attains at most 67.6% majority accuracy, below its parent's 69.6% single-sample score. Oracle headroom is neither reliably selected nor proof of valid reasoning, as illustrated by the answer-correct but derivation-invalid candidate in Appendix C.3. Conversely, a contracted candidate pool limits what a selector can extract from those recorded candidates.

Majority voting selects without the reference answer, and its accuracy is then graded. An oracle–majority gap concerns this selector, not every verifier-free or learned selector, which would require separate evaluation. Near zero same-model headroom, raw $\mathrm{pass@1}$, $\mathrm{pass@}K$ and $\mathrm{maj@}K$ are more informative than an unstable extraction ratio. Changes are paired with each checkpoint's own parent, not a pooled parent median.

B.9 From output length to cumulative token use and resource limits

For mean all-output length $\bar{t}_m$, k candidates per problem use $k\bar{t}_m$ tokens on average. At fixed k, this total changes proportionally with mean length, and varying the candidate count changes it further. Figures 3, 2 and 14 report these all-output measurements before target or correctness normalization. Length alone does not establish reasoning progress, and untruncated means describe selected outputs, not uncapped continuations.

Qwen expert-75%'s 246-sample oracle target entails 12.5 times the parent's generated-token work, $k^*_m(0.9)\bar{t}_m$. Even excluding prompt and verification, cost per generated token must fall below 8.0% of the parent's to break even. Expert-50% instead requires 22 versus 20 samples and 1.12 times the token work. These are conditional requirements, not hardware measurements. Fewer resident experts need not reduce the number of experts activated per token.

Figure 18 combines mean generated length $\bar{t}_m$ with single-sample correctness $\mu_m = P_m(1)$ in fractional units. Zero-success checkpoints have no finite observed cost per correct completion. For

$\mu_m > 0$, the pooled sampling-work proxy is

$$T_m = \frac{\bar{t}_m}{\mu_m}, \qquad \frac{T_m}{T_0} = \frac{\bar{t}_m}{\bar{t}_0}\frac{\mu_0}{\mu_m}. \tag{7}$$

The factorization is exact for the defined aggregate. It counts generated tokens per correct completion across the evaluated mixture of problems. It is not $\mathbb{E}_q[t_q/p_q]$, the cost of independently solving each problem by repeated sampling, which may behave very differently when some p_q are very small. It also excludes prompt processing and verification cost.

Token counts are compared within a backbone, since tokenizers differ. Even within a backbone, a token ratio does not become a FLOP ratio after multiplying by one minus the nominal pruning rate. Removing experts primarily changes the resident pool, whereas routing can keep the number of activated experts fixed. FFN and depth removal change other operations. No complete latency or throughput measurement matrix is available for this grid, so we do not claim a hardware ranking among the axes.

All cost values refer to the capped generation policy. The cap censors how long an uncapped run might continue, but this fact alone does not establish a lower bound on uncapped cost per correct answer. Extra tokens might also change correctness. The numerator and denominator would both need to be measured under the alternative policy.

The proxy also inherits the grader's limitations. Gemma RAZOR at 50% retains much of Minerva's accepted-answer coverage despite lower single-sample and majority accuracy, but loses substantially more coverage on AIME. These task-dependent scores do not establish preserved valid reasoning at greater token cost.

C Problem- and Trajectory-Level Analyses

We pair problem identities to examine changes in observed solved sets and success frequencies, then separate accepted answers from valid derivations through a concrete counterexample. Trajectory-level measurements describe token use, repetition and surface diversity without certifying useful self-correction or distinct valid strategies. Statistical references and robustness checks are collected in Appendix D.

C.1 Retained, lost and newly observed problems across pruning rates

Permuting success counts among equally sampled problems preserves the entire pass@k curve. We pair problem identities to distinguish retained, lost and newly observed successes. A lost success has at least one correct parent candidate but none in the child pool. This is an observed transition, not impossibility.

Figure 19 pairs all core Qwen rates. On AIME, expert-50% and expert-75% solve the parent's same 29 of 30 problems within 2,048 samples. One expert-75% problem has only a single success, so set equality does not establish saturation. FFN-40% instead retains 19 problems and loses ten, while depth-20% retains 24 and loses five. Neither adds a new success. Both show contraction of the observed parent-solved set.

The common 2,048-candidate Minerva pools expose replacement. At expert-50%, both models solve 113 problems, but the child retains 108, loses five and newly solves five. At expert-75%, the 129-problem set comprises 107 retained and 22 newly observed successes, alongside six lost parent successes. Even higher coverage need not be a superset of the parent's observed range.

C.2 Success-frequency transitions

Figure 20 pairs the same 30 AIME problems between Qwen and three children, using 2,048 candidates per problem. Four success-frequency bands distinguish no observed success, positive frequency below 5%, 5% to below 50%, and at least 50%. These transitions separate lower reliability from loss of an observed success. "No hit" does not mean zero success probability.

Of the 22 problems solved on at least half of the parent's candidates, expert-50% leaves 21 in that band and moves one into the 5%–50% band. Under FFN-40%, only one remains in the highest

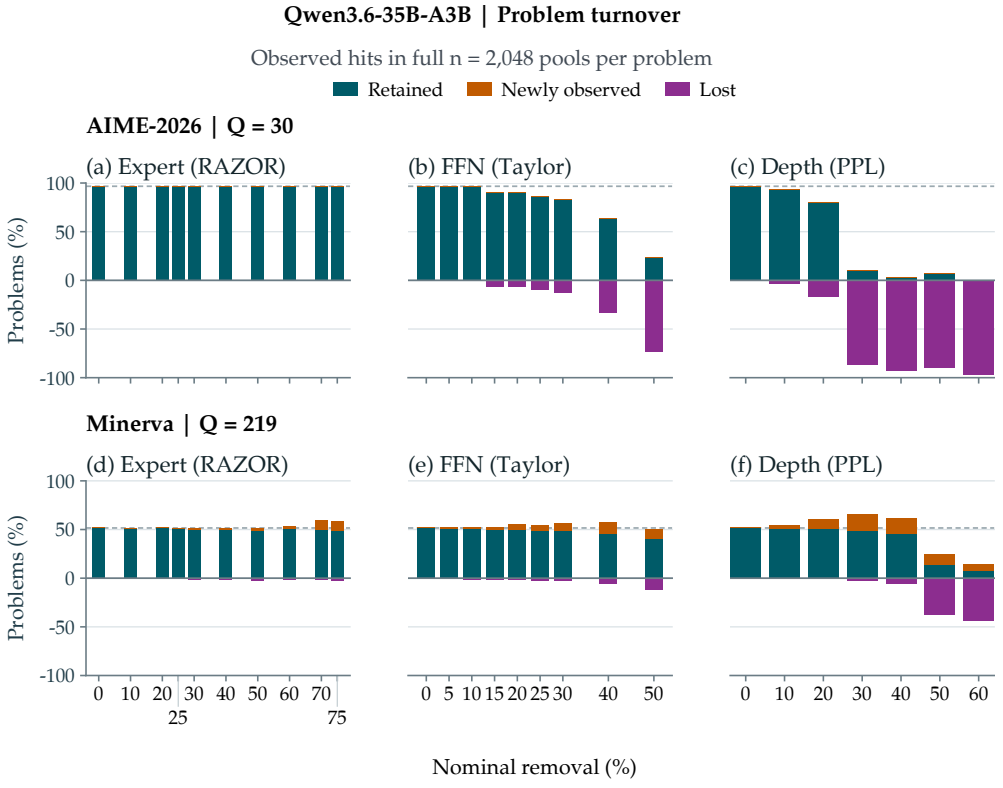


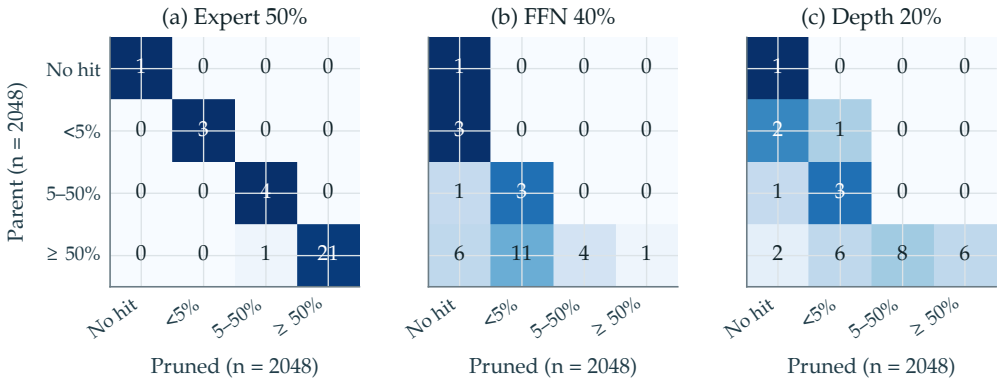
Figure 19: **Equal or higher coverage need not preserve the parent's solved set.** All core Qwen3.6-35B-A3B RAZOR, Taylor FFN and PPL depth rates are paired by problem identity at $n = K = 2048$. Positive stacks give child coverage. Retained plus lost equals parent coverage (dashed lines). Percentages use all 30 AIME or 219 common Minerva problems. New and lost refer to recorded pools, not proof that a reasoning capability was created or deleted.

band, four move into the middle band, eleven become rare successes below 5%, and six have no observed success. Depth-20% retains six in the highest band and loses observed successes on two of these 22 problems. These examples distinguish frequency loss from observed-set contraction without establishing a general severity law. Appendix D.2 asks whether frequencies can be reordered even when a solved set is retained.

Grader-accepted successes must also be distinguished from valid reasoning. For example, under the corrected 272-problem grader (Appendix A.4), Gemma RAZOR at nominal 30% increases Minerva's measured pass@2048 from 69.1% to 72.1%, while single-sample accuracy falls from 49.6% to 44.0%. Majority-vote accuracy has not yet been recomputed under this corrected grader. Under the original 219-problem grading it fell from 36.5% to 33.8%. This describes coverage under the stated grader, whose answer-format and parser limitations are discussed in Appendix E.2. A confidence interval cannot remove systematic grading error or certify the validity of the newly accepted derivations.

C.3 A CORRECT ANSWER WITH AN INVALID DERIVATION

In the historical Qwen expert-25% comparison (Appendix D.6), the rerun gains exactly one covered question, changing from zero to one accepted answer among 2,048 candidates. This raises its endpoint by 3.33 percentage points despite little change in single-sample accuracy. The sole successful candidate terminates normally with boxed answer 393. We examine it because it determines this coverage change, not as a random sample for estimating reasoning-error prevalence.



AIME-2026, paired by item ID. "No hit" means zero correct samples in this run.

Figure 20: **AIME success-frequency transitions.** Rows are parent bands and columns are child bands. Labels count problems, including empty transitions marked zero, and colour gives the fraction within each parent row.

The question counts ordered tuples over $\{1,2,3\}^7$ satisfying $\sum_i a_i \equiv 0$ and $\sum_{i=1}^{7} a_i a_{i+1} a_{i+3} \equiv 0 \pmod 3$, with cyclic indices. Exact integer enumeration of all 3^7 tuples gives 393 solutions. Among the 729 satisfying the first constraint, the cubic residues have counts $(393, 168, 168)$. The final answer is therefore correct, but the candidate's *derivation* contains an uncorrected error.

Writing $x_i = a_i \bmod 3$, the candidate defines $Z(u,1) = \sum_{\boldsymbol{x}} \omega^{u\sum_i x_i + S(\boldsymbol{x})}$, where $\omega^3 = 1$, $\omega \neq 1$ and S is the stated cubic. For $\boldsymbol{y} = (x_2, \ldots, x_7)$, set

$$A = x_2x_4 + x_5x_6 + x_7x_3, \qquad R = \sum_{i=2}^{7} x_i,$$
$$B = x_2x_3x_5 + x_3x_4x_6 + x_4x_5x_7 + x_6x_7x_2.$$

Summing out x_1 gives

$$Z(u,1) = 3 \sum_{\boldsymbol{y}: A \equiv -u} \omega^{B+uR}. \tag{8}$$

The candidate omits uR in the $u = 1, 2$ terms and repeats those equations in its final verification. This is not merely a discarded intermediate attempt. For example, with $u = 1$ and $\boldsymbol{y} = (1,1,2,0,0,0)$, the correct inner sum is 3ω, not 3. Exhaustive residue counting also falsifies the complete sums without floating-point approximations, giving

u	0	1	2
$Z(u,1)$ from its definition	567	−189	297
candidate's omitted-phase expression	567	54	54

The two errors, $+243$ and -243, cancel in the total. Both routes yield the final value 393, but the displayed derivation does not justify it.

This is an *answer-correct, derivation-invalid* example, not a false positive under the benchmark's final-answer criterion. We leave its original score unchanged. It demonstrates concretely why one newly accepted candidate can change large-budget coverage without certifying a valid reasoning trace. It does not estimate how often this occurs or show that pruning caused the error. The score-based calibration certificates and candidate-pool comparisons remain statements about the fixed grader, not proof-quality guarantees.

C.4 Token efficiency within and across trajectories

Token use and reasoning progress are separate questions. All-output token use describes the full benchmark, whereas paired comparisons condition on problem identity, correctness and termination.

Correct-output comparisons on jointly solved subsets require explicit inclusion rules, sample counts, equal-problem weighting and fixed-set sensitivity. They do not replace full-task error, truncation and token-use measurements.

Assessing useful verification or backtracking requires annotations of valid checks and corrections, with defined tokenization and reasoning/answer boundaries, rather than reflective keywords. These annotations remain incomplete. Cumulative sampling-token use depends on both mean length and candidate count. Neither it nor the proxies in Appendix B.9 establishes reasoning progress by itself.

Results. The process measurements cover 23 Qwen3.6-35B-A3B checkpoints and 26 GLM-4.7-Flash checkpoints, including each unpruned parent. Relative to their parents, incorrect complete traces are shorter in 11 of 12 testable cells and more repetitive in 11 of 12. The cells form the cross-product of two backbones, three pruning axes and two mathematics tasks, AIME-2026 and Minerva. These counts compare the deepest eligible recorded rate with its parent, not every adjacent pair of rates. Figure 21 shows the AIME-2026 sweeps. Length and repetition can change differently. Qwen expert removal reduces failing-trace length from 2233 to 963 whitespace tokens at 75%, while 4-gram repetition is 0.076 against the parent value of 0.078. GLM depth removal instead reaches 3128 tokens and 0.326 repetition at 19%.

On AIME-2026, FFN removal has a positive rank association with 4-gram repetition of incorrect complete traces in both backbones. Spearman correlation is $\rho = 0.98$ ($p < 0.001$, $n = 8$) for Qwen3.6 and $\rho = 0.79$ ($p = 0.048$, $n = 7$) for GLM-4.7. The exact permutation p-values describe the observed rate grid, not a population of independently sampled checkpoints.

Figure 22 compares truncation with the number of problems solved by at least one complete trace. Truncation peaks before the deepest measured tier and then falls in 3 of 12 cells, all depth sweeps. On AIME-2026, Qwen depth removal peaks at 91.0% truncation at 30% removal and falls to 54.5% at 50%. GLM depth removal falls from 100.0% to 92.7%. Complete-trace coverage remains low at these deepest tiers. Lower truncation therefore coexists with short repetitive failures rather than indicating recovered accuracy (Appendix B.7).

Shaded spans mark tiers with no problem solved by a complete trace. GLM depth-38, for example, averages 73 whitespace tokens despite solving no problems. The endpoint comparisons and reported trend statistics stop at the last eligible rate with a solved problem. Trace-length and repetition means condition on complete outputs and do not describe truncated generations.

What remains open. Establishing effective verification or correction requires the annotations described above. Reflection cues, length and repetition alone do not establish reasoning progress. Paired correct-only length and repetition comparisons describe jointly solved subsets, but cannot replace these annotations. Such comparisons are undefined where the jointly solved subset is empty.

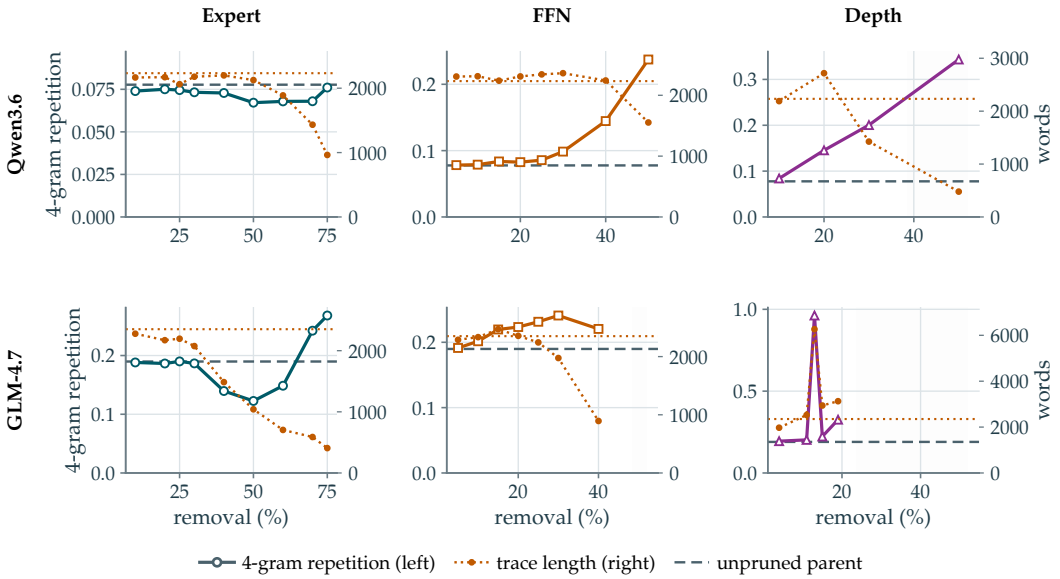


Figure 21: **Length and repetition of incorrect complete traces.** Rows are backbones and columns are pruning axes on AIME-2026. Solid lines use the left axis for 4-gram repetition. Dotted lines use the right axis for length in whitespace tokens of saved text, not serving-tokenizer counts. Dashed black lines mark the unpruned parent. Shading marks tiers with no problem solved by a complete trace. The displayed trajectories need not change monotonically with removal. Repetition does not establish reasoning validity. Candidate pools contain 128 samples per problem, rather than the 2,048 used in the main comparison, and all trace means condition on complete outputs.

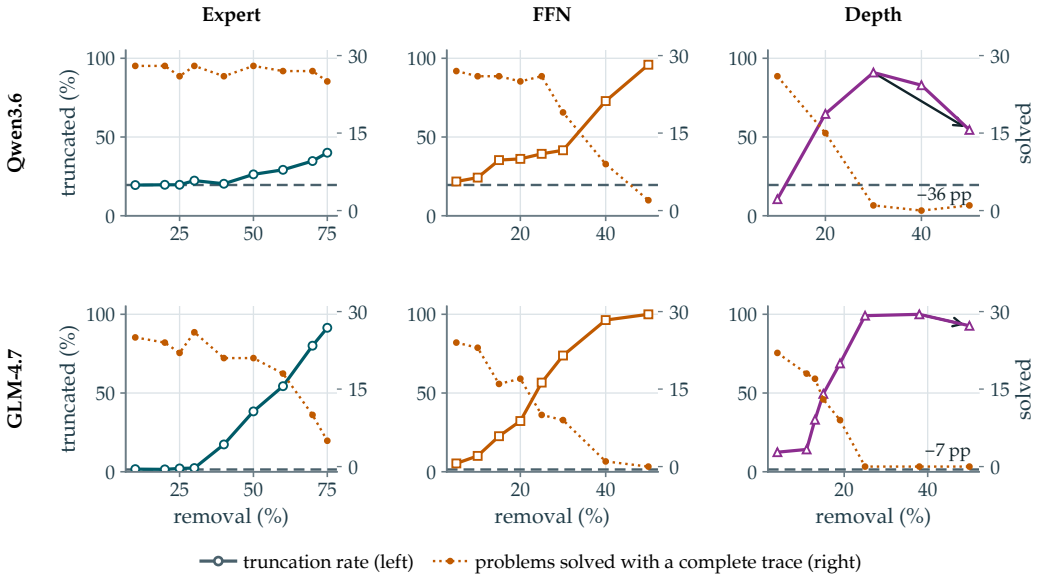


Figure 22: **Truncation and solved-problem counts.** Solid lines use the left axis for truncation. Dotted lines use the right axis for AIME-2026 problems solved by at least one complete trace. Red arrows mark depth-axis reversals where lower truncation does not imply recovered coverage, consistent with Appendix B.7.

C.5 DIVERSITY OF VALID SOLUTION STRATEGIES

Valid-strategy diversity concerns substantive methods, such as theorems or algorithms, rather than wording. Matching total-candidate counts measures discovery jointly with success frequency, whereas matching correct-sample counts conditions on sufficient successes. Both require per-problem sample availability, inclusion criteria and fixed-set sensitivity. Method-level grouping and manual validation are needed to distinguish valid approaches from incorrect paths. Repetition and our cross-candidate 4-gram overlap (a symmetric clipped-overlap statistic, not standard Self-BLEU) remain auxiliary measures and do not by themselves establish strategy diversity.

Surface-form results. We measure lexical overlap and answer position in saved trajectories. These surface statistics complement, but do not replace, semantic grouping of validated solution methods.

At the deepest eligible comparison rate, cross-candidate 4-gram overlap is no higher than the parent in all 12 testable backbone–axis–task cells. This endpoint comparison does not imply a monotonic decline or exclude intermediate values above the parent (Figure 23, left). On AIME-2026, Qwen3.6 expert removal changes overlap from 0.146 at the parent to 0.105 at 75%, and GLM-4.7 from 0.152 to 0.123. Lower shared n-gram overlap does not establish a wider range of valid strategies. Two candidates may share no 4-gram while executing the same algorithm.

For a correct complete trace of L characters, let a be the character index of its first literal answer occurrence and b the start of its final $\backslash\texttt{boxed}$ or $\backslash\texttt{fbox}$ marker. The answer-to-box gap is $\max(0, b - a)/L$, defined only when both are present. It is zero when the first occurrence is inside or after the final marker and does not count answer occurrences. The last displayed depth points in Figure 23 (right) reach 0.251 at 20% for Qwen3.6 and 0.143 at 19% for GLM-4.7, above their respective parent values of 0.131 and 0.030.

A wider gap can reflect verification, redundant restatement or chance recurrence of a numeric string. The statistic does not distinguish these possibilities. Answers shorter than two characters are excluded. A separate three-character sensitivity check concerns first-answer position, not the answer-to-box gap. Neither measurement rules out literal-match artefacts. Cells with fewer than 100 correct complete traces containing a literal answer match are omitted. This excludes 7 checkpoints, including GLM-4.7 depth-13. Of its 514 correct traces, 509 are complete. Among the complete traces, 507 lack a literal answer match and the remaining 2 have no box marker, leaving no usable gap. For the same checkpoint, 99.4% of correct complete traces contain looping, with mean 4-gram repetition of 0.936, while complete traces solve 17 of 30 AIME-2026 problems. Surface repetition can therefore coexist with accepted answers.

What remains open. Claims about valid-strategy diversity require method-level grouping, manual validation, annotation agreement and comparisons with matched correct-sample counts. These semantic audits remain incomplete. Repetition, the custom symmetric clipped 4-gram overlap reported here (not standard Self-BLEU), and answer position are auxiliary measures. For each problem, overlap uses all pairs among the first up to 16 complete outputs encountered in stored generation order. Problems with fewer than two complete outputs are undefined.

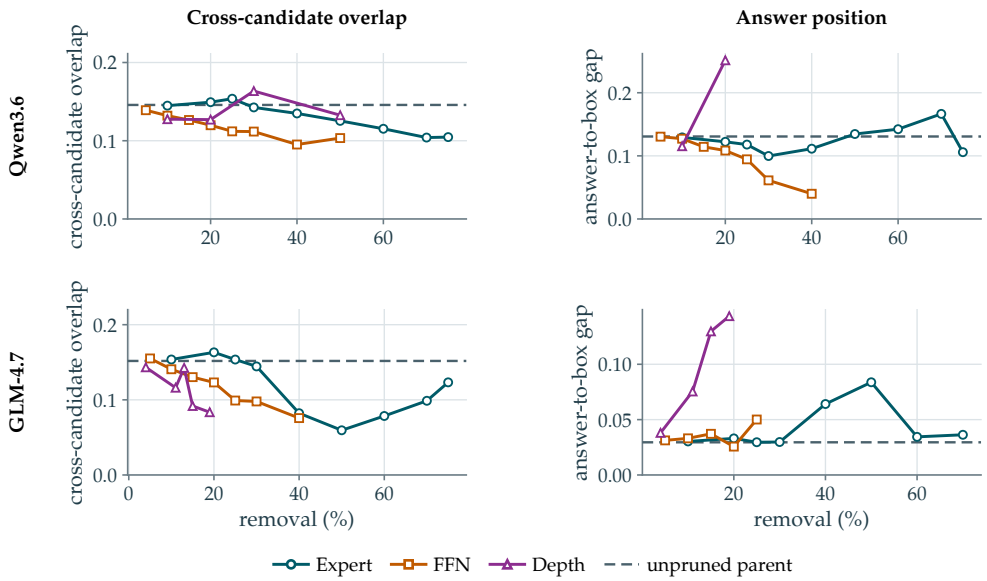


Figure 23: **Surface diversity and answer position on AIME-2026.** Left panels show mean cross-candidate 4-gram overlap for the same problem. Right panels show the mean answer-to-box gap among correct complete traces with both a literal answer match and a box marker. Neither statistic measures semantic strategy diversity. Answer-position cells with fewer than 100 correct complete traces containing a literal answer match are omitted. The underlying pools contain 128 outputs per problem, and overlap uses the first up to 16 complete outputs in stored order and is defined for problems with at least two such outputs.

D Statistical Support and Robustness

This appendix collects statistical support for the model- and problem-level comparisons. Uniform-loss references and difficulty-ordering checks address distinct explanations of the paired outcomes. We then report paired-problem uncertainty, checkpoint matching, extended sampling, decoding controls and same-seed run sensitivity. The remaining subsections give the supporting calculations, sampling assumptions and mathematical foundations. These analyses do not provide independent-seed replication or introduce a new pruning method.

D.1 Can uniform success loss explain the coverage changes?

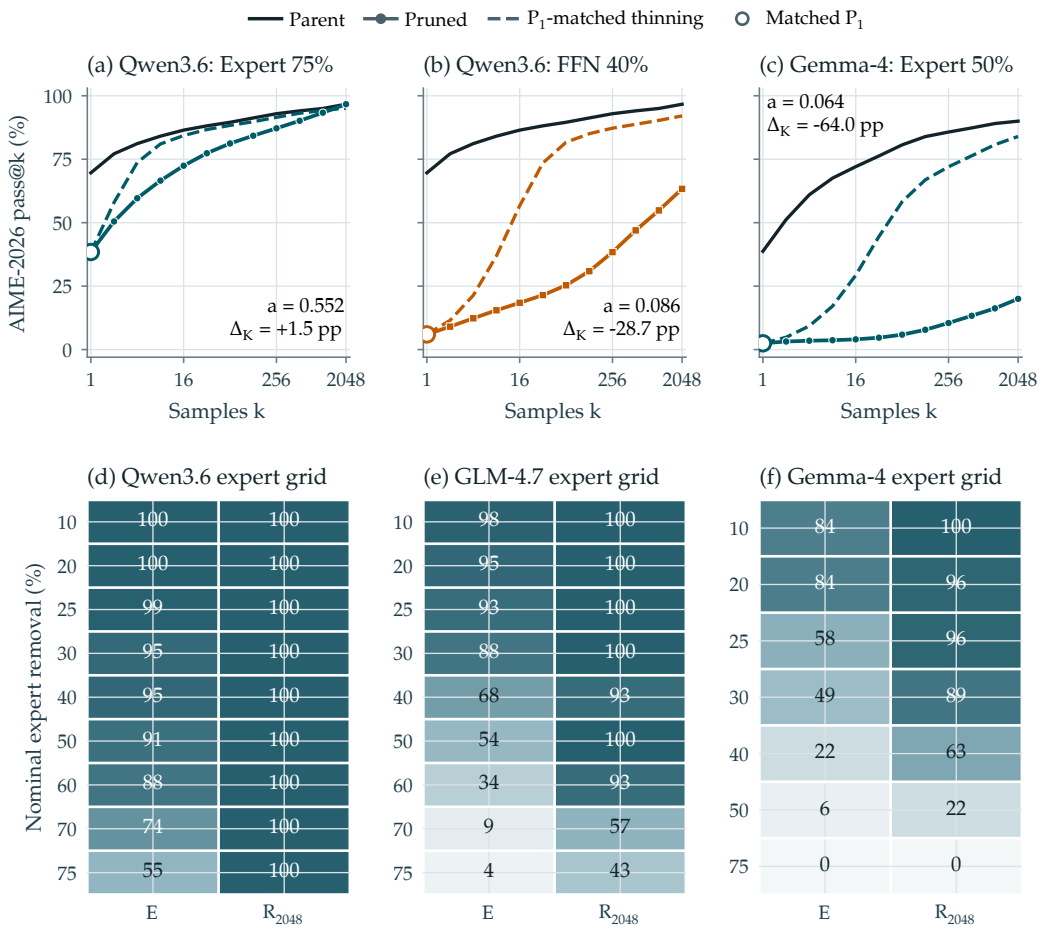
Rather than matching two checkpoints on their aggregate scores, the uniform-loss reference matches the child's mean accuracy in expectation within the parent's recorded pool. We first define that comparison and report its observed deviations. Appendix D.7 then examines candidate-block transport and sensitivity to the mean-matching rule. Appendix D.9.1 supplies the finite-pool identities, and no population probability model is needed for this conditional calculation.

D.1.1 A rate-matched uniform-loss reference

A lower mean alone can depress the entire curve. Our conditional reference retains each recorded parent success independently with probability $a_m = P_m(1)/P_0(1)$, leaving failures and problem weights unchanged. For $0 \le a_m \le 1$, this matches the child's mean score *in expectation* without fitting its higher-budget coverage.

For a random k-candidate subset on problem q, the number of original successes is $J_{qk} \sim \mathrm{Hypergeom}(n_{0q}, c_{0q}, k)$. The reference coverage is

$$U_m(k) = \frac{1}{|\mathcal{Q}|} \sum_q \left[1 - \mathbb{E}\left[(1-a_m)^{J_{qk}}\right]\right], \qquad D_m(k) = P_m(k) - U_m(k). \tag{9}$$



AIME, K = n = 2048. Grid entries: retention (%); each backbone uses its own parent.

Figure 24: **Rate-matched references and parent-normalized expert sweeps.** Upper panels show selected AIME pass@k curves. Dashed curves are uniform-loss references, with a denoting the success-retention rate and Δ_K the observed-minus-reference coverage. Lower panels place nominal removal rates in rows and E, R_{2048} in columns, retaining nine Qwen and GLM rates and seven Gemma rates. E is single-sample accuracy retention, not chain-of-thought efficiency. R_{2048} is coverage retention relative to the same parent. These diagnostics supplement the full primary sweeps in Figure 1 without determining checkpoint selection.

When all $n_{0q} = K$, the reference endpoint is simply $U_m(K) = |\mathcal{Q}|^{-1} \sum_q [1 - (1 - a_m)^{c_{0q}}]$. Negative $D_m(k)$ is excess coverage loss relative to uniform thinning, not relative to every possible transformation of success probabilities. An all-zero child gives $a_m = 0$ and contains no shape information for this test. An empty parent or $a_m > 1$ is outside its scope.

D.1.2 Observed deviations from uniform loss

This comparison uses $K = n = 2048$ for AIME and $K = n = 512$ for HumanEval+. LCB comparisons separate equal pools at $K = n = 256$, equal pools at $K = n = 512$, and 512-candidate parents paired with 256-candidate children at $K = 256$. AIME intervals use 2,000 paired-problem resamples, re-estimating the thinning rate and conditioning on admissible replicates (Appendix D.7). Qwen3.6 and Gemma-4 in the row labels abbreviate Qwen3.6-35B-A3B and Gemma-4-26B-A4B-it.

On AIME, Qwen expert pruning at 75% reaches 96.7% coverage, close to the uniform-thinning reference of 95.2%. Qwen FFN pruning at 40% instead reaches 63.3% against 92.1%, a −28.7-point

residual with a paired-problem 95% interval of $[-43.7, -14.1]$. Gemma expert pruning at 50% reaches 20.0% against 84.0%. Unlike a caliper match between two checkpoints, each comparison holds mean accuracy fixed in expectation.

The code tasks make the scope of this effect clearer. At $K = 256$ on LiveCodeBench, Qwen FFN-40% has a -46.0-point residual. No bootstrap interval is reported for this unequal-pool comparison. Gemma expert-50% has a -45.7-point residual $[-51.9, -38.7]$. Their HumanEval+ residuals are only -8.7 and -5.8 points. These examples sit among the applicable calibrated primary checkpoints. Zero-success cases, including 12 of the 32 calibrated depth checkpoints on AIME, are marked uninformative for shape rather than counted as successful reference predictions. Adding the tail-drop screens back raises that depth count to 25 of 50, which is the count the reference cohort summaries report.

Appendix D.7 tests two sensitivities of this comparison. Candidate-block transport asks whether a rate fitted on one block predicts another block of the same problems. Alternative scales ask how the result depends on the rule used to match mean accuracy. Neither check tests all possible scalar explanations, and Appendix D.2 addresses the separate problem-order question.

D.2 Does pruning preserve the ordering of problem difficulty?

The next question goes beyond any particular rate-matched reference to ask whether a common nondecreasing mapping can explain the paired success probabilities. Solved-set labels alone do not answer this probability-order question, and absent certificates do not establish preserved ordering.

D.2.1 Similar curves and changed difficulty

Figure 25(a,b) shows all 30 problem frequencies and full-pool Clopper–Pearson (CP) rectangles for an ID-ordered certified pair. Worst-item bounds use all certified pairs, not only the illustrated pair. Panel (c) compares the 99 calibrated checkpoints over every integer $1 \le k \le 2048$, and its horizontal scale is linear up to one point and logarithmic thereafter. The 18 tail-drop screens remain in the reference cohort summaries but are omitted from this display. IID and chunk bounds belong to separate confidence families. The 3 outcome-selected rows remain descriptive, and zero bounds do not establish equivalence.

Aggregate curve agreement does not identify problem ordering (Appendix D.9.2). To test whether parent difficulty remains informative up to a common calibration, we examine $p_{mq} = f_m(p_{0q})$ for any nondecreasing f_m, without restricting its shape or matching means. A pair with $p_{0i} > p_{0j}$ but $p_{mi} < p_{mj}$ excludes all such transformations.

To distinguish reversals from sampling noise, let $[L_{tq}, U_{tq}]$ be simultaneous success-probability intervals. A certificate requires

$$L_{0i} > U_{0j} \quad \text{and} \quad U_{mi} < L_{mj}. \tag{10}$$

Two-sided Clopper–Pearson intervals use a Bonferroni correction over 11700 slots, counting shared parents once and reserving all three windows for screened-out checkpoints. For prespecified runs, within-stream IID Bernoulli sampling at fixed indices bounds the probability of any false certificate in this family by 5%, without independence across problems or overlapping blocks. This guarantee does not adjust for retrospective run adoption. Split agreement does not establish IID.

Among 114 full-budget primary checkpoints eligible for this inference, 48 have at least one full-pool certificate. The other 3 outcome-selected checkpoints remain in descriptive analyses but are excluded from certificate and stability summaries. For Qwen FFN-5%, the maximum absolute residual over *all integer budgets* $1 \le k \le 2048$ is only 0.94 points, yet 5 certified reversed pairs involve 7 problems. Every common nondecreasing calibration must err by at least 5.77 points on some problem. A certified child reversal of size g forces at least one endpoint error of $g/2$. Disjoint pairs also lower-bound mean absolute error (Appendix D.8.2). This exploratory example has no pair certified in all three blocks. Gemma expert-20% has 3.25 points of maximum all-budget deviation and 30 certified pairs over 17 problems, including 2 pairs present in all three blocks. Its worst-problem calibration error is at least 25.50 points. Gemma does not meet the FFN example's one-point criterion. Overlapping pairs and blocks are not independent replications, and absent certificates are inconclusive.

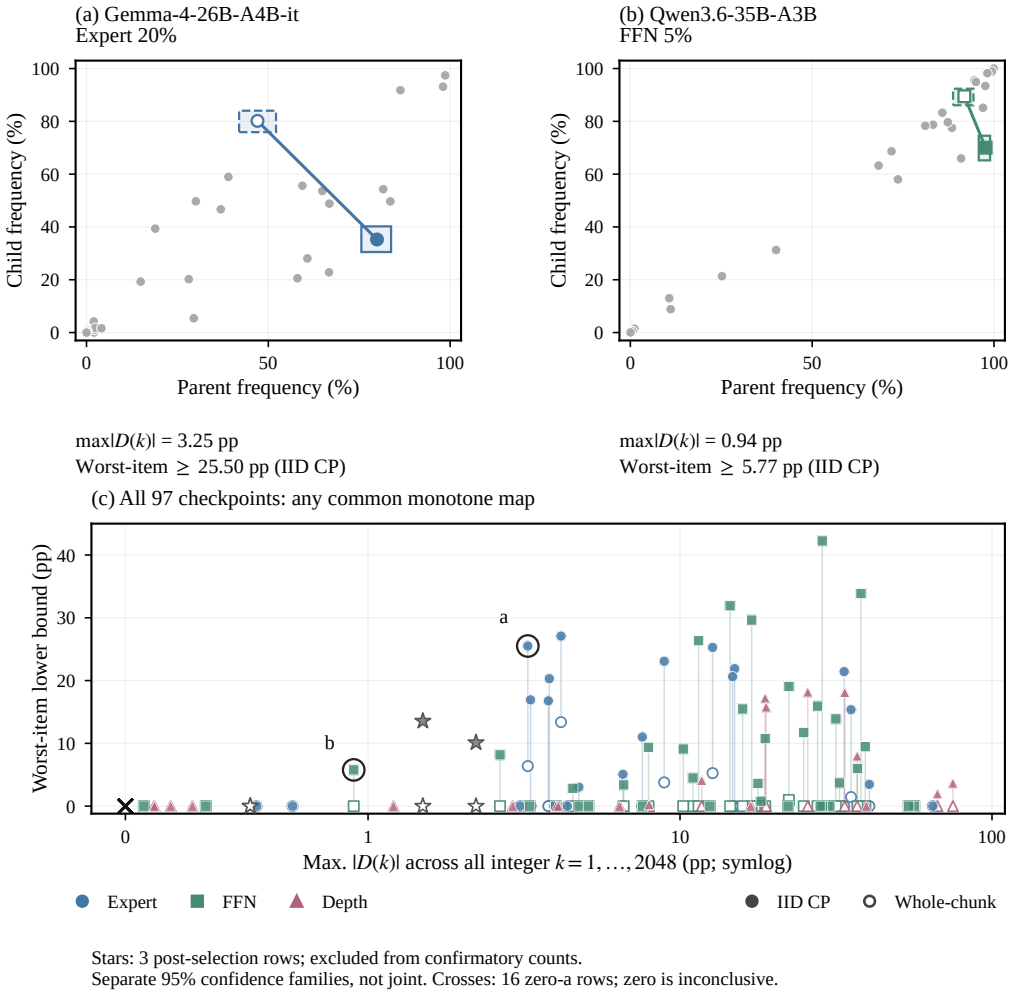


Figure 25: **Similar scaling curves need not preserve problem difficulty.** Panels (a) and (b) show problem frequencies and CP rectangles, where fill distinguishes problems and segments are not fits. (c) Reference deviation versus worst-item mismatch. Filled and open marks denote IID and chunk bounds, and stars denote outcome-selected rows. Tail-drop is omitted from panel (c). Vertical connectors are not error bars, and zero-retention crosses coincide.

Relaxing candidate independence. Treating each 32-candidate request as one bounded observation permits arbitrary within-request dependence. A method-adjusted betting calculation retains certificates for 6 checkpoints, including 2 pairs for Gemma expert-20% and a 6.37-point worst-problem calibration-error lower bound. GLM expert-50% and Qwen FFN-5% no longer yield a certificate. It assumes constant conditional means across chunks, not unrestricted cross-chunk dependence. The eight-chunk prefix is structurally too short for this certificate. Appendix D.8.5 reports the complete cohort and the separate error budget, rather than interpreting a failed certificate as evidence for unchanged ordering.

D.3 Paired-problem uncertainty

We jointly resample problem IDs for a parent and its pruned checkpoints, recomputing every metric on the same IDs to preserve pairing and dependence among budgets. Ratios with zero resampled denominators and unattained targets are counted as undefined or censored, not silently discarded. Each interval reports the number of resamples used.

Problem bootstrap intervals concern the sampled task population conditional on the observed candidates. They do not estimate training-seed variation or correct systematic grading errors. In particular, resampling an all-zero success vector cannot establish that its underlying success probability is zero.

D.4 Extending Minerva sampling beyond 2,048 candidates

The extended evaluation retains the original candidates and adds 2,048 per problem for each core parent and the expert-10% through expert-50% checkpoints. Two quantities must be distinguished. The change in observed solved counts compares the nested 2,048- and 4,096-candidate pools. In contrast, the budget gain compares $k = 2048$ and $k = 4096$ within the same complete 4,096-candidate pool. The latter re-estimates all lower-budget scores and need not equal the percentage of newly solved problems. These are nested observations, not independent repetitions. Severity, solved-set comparisons and the broader mathematical diagnostics continue to use the common 2,048-candidate pools. Neither endpoint establishes exhaustion of the model's possible successful outputs.

Thirteen of the fourteen extended pools solve one to six additional problems. The Qwen and GLM parents add two and one, respectively. GLM expert-25% adds none, although its same-pool budget gain is 0.81 percentage points. This difference matters because a larger random subset can be more likely to include a known success even when the added candidates solve no new problem. A separate extension of Qwen depth-60% raises the observed solved count from 32 to 46 of 219, adding 14 problems. Its complete 4,096-candidate pool estimates pass@2048 at 16.66% and pass@4096 at 21.00%. The same-pool gain is 4.35 percentage points. Figure 1 uses this extended pool but displays only $k \leq 2048$. This depth checkpoint is separate from the fourteen parent/expert comparisons below. Figure 26 compares the old and extended observed coverage with pass@2048 re-estimated from the complete pool. Four EAN and routing-frequency controls are shown separately from the fourteen primary comparisons, and all use the same 219 problems.

D.5 Sensitivity to decoding and trajectory-length limits

Pruning may change how well a fixed decoder suits the model. The available controls vary output caps for Qwen and repetition penalties for unpruned parents, leaving broader decoder robustness unresolved.

Temperature and replication. A controlled temperature comparison requires matched candidate counts and length limits, fixed problem and checkpoint pairs, and independent seeds, without test-set tuning. Full pass@k curves and problem transitions would then distinguish coverage from success frequency. Reused blocks and same-seed reruns are not independent replications.

Single-trajectory budget. Thinking-duration control (Muennighoff et al., 2025) differs from a total-output cap. Isolating it requires reserved answer tokens and explicit stopping or continuation rules, alongside actual length, truncation and missing-answer rates. Truncation summaries (Appendix B.7) motivate this untested comparison but do not show that longer reasoning improves coverage or validity.

Output-cap comparisons. The change from an 8,192-token to a 16,384-token output cap has different outcomes across the tested Qwen checkpoints (Figure 27, top). On AIME-2026 with 2,048 candidates per problem, expert-30 gains 7.70 percentage points in pass@1, from 65.92% to 73.62%, while truncation falls from 22.8% to 0.1%. For depth-30, the gain rounds to 0.00 percentage points and truncation remains high at 93.0% versus 92.9% under the original cap. These observations concern the two tested caps on one task. They do not identify truncation as the sole constraint or predict responses to larger budgets.

Repetition-penalty sensitivity. The unpruned parents respond differently to the tested repetition penalties on AIME-2026 (Figure 27, bottom). Qwen3.6 pass@1 ranges from 69.45% to 70.22% across penalties 1.00 to 1.15. GLM-4.7 pass@1 falls from 32.53% to 25.14% at 1.05 and to 0.00% at 1.10, where truncation rounds to 100.0%. This parent-level sensitivity does not establish how pruning changes the response to a repetition penalty.

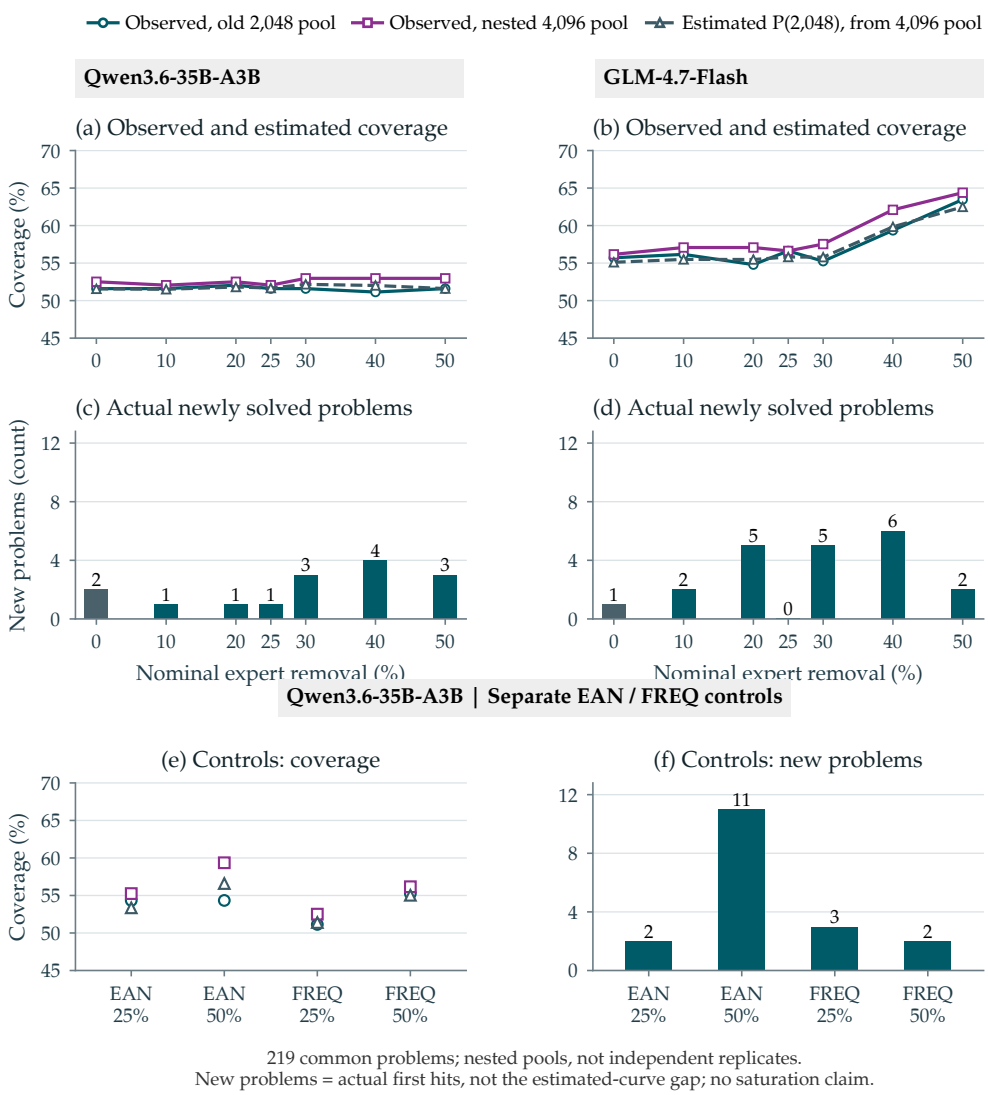


Figure 26: **Minerva coverage beyond 2,048 candidates.** Panels (a) and (b) show coverage across expert-removal rates, panels (c) and (d) show newly solved problem counts, and panels (e) and (f) show separate EAN and routing-frequency controls. Bars count new problems, not same-pool probability gains.

What remains open. The output-cap comparison covers only Qwen, with no GLM 16k pool, and the penalty sweep covers only unpruned parents. Temperature sweeps and independent-seed replication remain unavailable. The available reruns reuse the same seed. Total-output caps do not control thinking duration separately from answer generation. A thinking-duration intervention with reserved answer tokens remains untested.

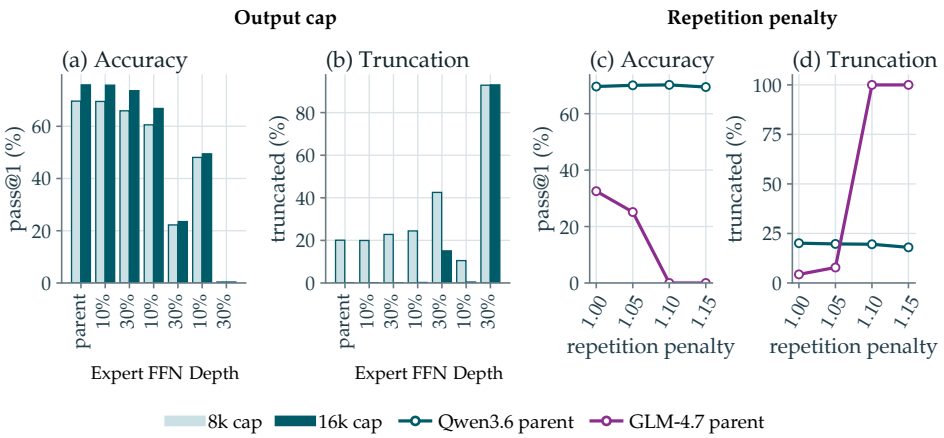


Figure 27: **Output cap and repetition penalty.** Top panels show AIME-2026 pass@1 and truncation under 8k and 16k output caps for Qwen. Bars are grouped by pruning axis and labelled by removal fraction. The unpruned parent is omitted because its 8k summary lacks truncation data. The depth-30 accuracy bars are near zero, with pass@1 of 0.37% under either cap. Bottom panels show both quantities for the two unpruned parents across repetition penalties. No pruned checkpoints enter the penalty sweep, so it does not measure pruning-induced decoding sensitivity.

D.6 Same-seed run sensitivity

Three existing Qwen expert checkpoints at 25%, 30% and 40% removal were reevaluated on AIME using the same base seed 42 and 2,048 candidates per question. Each rerun generated a separate pool rather than extending the original one. Indexed request seeds and sampling parameters agree where recorded. The earlier runs evaluated both AIME and Minerva, whereas the reruns evaluated AIME alone. Historical runtime settings are incomplete. Differences between these runs therefore cannot be attributed to a single serving-stack change or interpreted as independent-seed variation. They also do not replicate the FFN-5% finding, whose independent-seed limitation is discussed in Appendix E.5.

The audit compares single-sample accuracy and observed coverage under the common grader. Existing rescoring reports indicate no changed binary correctness labels across the six runs, but the grader has not been independently rerun for this analysis. Unchanged binary labels also do not establish unchanged normalized-answer groups. Standardizing candidate ordering and Monte Carlo draws can change intermediate majority scores, although all six majority-vote endpoints are 86.67%.

Only 9,720, 6,071 and 50 of the 61,440 paired candidate positions contain exactly identical text for the 25%, 30% and 40% checkpoints, respectively. These correspond to 15.82%, 9.88% and 0.08% exact agreement. Thus even matched seed labels do not yield identical trajectories under the incompletely recorded runtime conditions. This audit reports historical run sensitivity, not the final pool selection for the primary analysis. The comparisons remain descriptive and provide no independent-seed replication or explanation of the runtime difference.

Leverage of a unique success. For a fixed n-candidate pool, changing one successful label to failure on a question with $c \geq 1$ successes reduces aggregate coverage by

$$\delta P(k) = \frac{1}{Q} \frac{\binom{n-c}{k-1}}{\binom{n}{k}}.$$

This follows from Pascal's identity. If $c = 1$, the change is $k/(Qn)$. At $Q = 30, n = 2048$, a unique success contributes only 0.00163 percentage points to $P(1)$ but 3.33 points to $P(2048)$. This is a conditional influence calculation, not a relabeling of the observed data or an attribution of all between-run curve differences to one candidate.

D.7 TECHNICAL SUPPORT FOR FINITE-POOL REFERENCES AND SENSITIVITY

Appendices D.1 and D.2 use two different references, a mean-matched loss of recorded successes and an ordering constraint on problem-level probabilities. Their calculations and assumptions are separated here so that rejecting one is not mistaken for rejecting every scalar account.

D.7.1 FACTORIAL-MOMENT INTERPRETATION

The diagnostic has an exact finite-pool interpretation. Write $M_{m,j} = |\mathcal{Q}|^{-1} \sum_q (c_{mq})_j/(n_{mq})_j$, where $(z)_j$ is a falling factorial. For $k \le \min_{t\in\{0,m\},q} n_{tq}$, inclusion–exclusion gives

$$P_m(k) = \sum_{j=1}^{k} (-1)^{j+1} \binom{k}{j} M_{m,j}, \qquad D_m(2) = -\big(M_{m,2} - a_m^2 M_{0,2}\big). \tag{11}$$

Thinning scales the jth factorial moment by a_m^j in expectation, so residuals probe allocation beyond the matched mean. At equal pool sizes and total successes, balancing counts weakly increases coverage at every budget, but residuals are not a complete concentration measure. Appendix D.9.1 proves these results. We compute Eq. 9, not the alternating sum or an IID approximation.

Paired-problem bootstrap intervals re-estimate a_m per resample, quantifying problem-set variation conditional on the pools, not a thinning randomization test. Mathematics checks fit on 256 candidates and evaluate on the remaining 1,792, then reverse the split and record held-out first-attempt mismatch. These test same-problem transport across blocks, not unseen-problem generalization.

D.7.2 FINITE-POOL CALCULATION AND CANDIDATE-BLOCK TRANSPORT

For a fixed parent pool, let $B_q \sim \mathrm{Binom}(c_{0q}, a_m)$ be the number of successes remaining after thinning. Averaging the finite-sample coverage estimator over B_q gives the same result as Eq. 9. Equivalently, we average $(1 - a_m)^j$ over the hypergeometric count of original successes in the selected subset. The operation marks successes as failures rather than deleting candidates from the pool, so the candidate budget remains unchanged.

The domain of the reference has to be separated from its performance. A child with higher single-sample accuracy than the parent requires $a_m > 1$ and is outside a success-retention model. Such a case is not clipped to one. A child with no successes yields zero reference coverage at every budget, so agreement there cannot be read as evidence for uniform damage. Parent problems with no observed successes also remain at zero in the reference. Newly observed child successes can therefore reflect the independent candidate pools rather than an expanded population support.

Bootstrap intervals use 2,000 paired-problem resamples with seed 20260911 and re-estimate a_m in each resample. The intervals concern a statistic of the recorded candidates. They are not calibrated tail probabilities under a sampled thinning null and do not include a fresh generation layer. Replicates with $a_m > 1$ or a zero parent denominator are recorded as undefined. Their percentile intervals are conditional on admissible replicates, which matters especially for near-parent checkpoints. Near-zero child rates can also yield a nontrivial mass of degenerate $a_m = 0$ resamples. These counts accompany each interval in the analysis output.

The first eight 32-sample chunks supply the 256-candidate block, and subtraction from the validated full counts supplies the disjoint 1,792-candidate block. The same problem IDs are used in both directions. Held-out single-sample mismatch is assessed without recalibrating on the test block.

D.7.3 ALTERNATIVE RATE-MATCHED SCALES IN THE SAME FINITE POOL

For an empirical parent frequency $\widehat{p}_q = c_{0q}/n_{0q}$, consider

$$f_b^{\mathrm{odds}}(p) = \frac{bp}{1 - p + bp}, \qquad f_b^{\mathrm{hazard}}(p) = 1 - (1 - p)^b, \qquad 0 < b \le 1. \tag{12}$$

We fit only b so that the mean transformed frequency equals the observed child mean. We then retain each parent success on problem q with probability $a_q = f_b(\widehat{p}_q)/\widehat{p}_q$ when $\widehat{p}_q > 0$ and use the same hypergeometric expectation as Eq. 9. Zero-success parent pools remain at zero. This construction

changes the allocation rule, not the candidate budget or the coverage estimator. It does not substitute $1-(1-f_b(\widehat{p}_q))^k$ for finite-pool coverage.

Both transformations leave $\widehat{p}_q = 1$ unchanged for every $b > 0$. Consequently, the proportion of full-count parent problems is a floor on the achievable mean. A target below that floor is unattainable, and a target equal to the floor is a limiting boundary unless the identity already attains it. Neither case is repaired by pseudocounts or clipping. The all-zero target is recorded separately as the degenerate operation retaining no successes, not as the continuous limit of a transformation at a full-count problem. A target above the parent mean is outside these loss-only families.

The endpoint-only sensitivity analysis retains every primary AIME, HumanEval+, and LiveCodeBench comparison with equal pool sizes, evaluated at $K = n$. LiveCodeBench pairs with unequal pools remain outside it, although their common-K uniform-thinning estimates are still computed. For the three candidate-block examples, it also fits b on one block and applies that same parameter to the other block's parent frequencies. Held-out first-attempt mismatch is retained. These checks probe transport of the chosen scale, not all monotone transformations and not generalization to new problems.

D.8 Technical support for ordering certificates and sampling assumptions

The ordering results in Appendix D.2 require simultaneous coverage under a stated sampling model. The following proofs give the deterministic implications of that coverage event. The subsequent sensitivity checks examine budgets, confidence families and within-request dependence.

D.8.1 Certificate validity

Proposition 1 (Excluding a common nondecreasing transformation)**.** *Let p_{tq} be a checkpoint's probability of a grader-correct completion on problem q under a fixed evaluation policy. Suppose intervals $[L_{tq}^{(b)}, U_{tq}^{(b)}]$ simultaneously cover these probabilities. If a parent–child pair and two problems satisfy $L_{0i}^{(b)} > U_{0j}^{(b)}$ and $U_{mi}^{(b)} < L_{mj}^{(b)}$, no single nondecreasing function f_m can satisfy $p_{mq} = f_m(p_{0q})$ for all problems.*

Proof. Simultaneous coverage implies $p_{0i} > p_{0j}$ and $p_{mi} < p_{mj}$. A nondecreasing function would require the opposite weak ordering for the child probabilities, a contradiction. □

D.8.2 How far from any common monotone calibration?

Excluding an exact transformation need not imply a large discrepancy. We therefore quantify a lower bound on approximation error. Let $\mathcal{F}$ contain all nondecreasing functions $f : [0,1] \to [0,1]$, and define, on the fixed $Q = 30$ evaluated problems,

$$\Delta_\infty = \inf_{f\in\mathcal{F}} \max_q |p_{mq} - f(p_{0q})|, \qquad \Delta_1 = \inf_{f\in\mathcal{F}} \frac{1}{Q}\sum_q |p_{mq} - f(p_{0q})|.$$

These are distances to a class of probability calibrations, not decreases in accuracy or regret for a sampling policy.

Proposition 2 (Simultaneous lower bounds on monotone approximation)**.** *On the simultaneous interval-coverage event, let $\mathcal{E}$ contain the certified pairs in Eq. 10, with directed edge (i,j) weighted by $g_{ij} = L_{mj} - U_{mi} > 0$. Then*

$$\Delta_\infty \geq \frac{1}{2}\max_{(i,j)\in\mathcal{E}} g_{ij}, \qquad \Delta_1 \geq \frac{1}{Q}\max_{\mathcal{M}\text{ a matching in }\mathcal{E}} \sum_{(i,j)\in\mathcal{M}} g_{ij}. \tag{13}$$

A matching uses each problem at most once. Empty maxima are defined as zero.

Proof. Fix any $f \in \mathcal{F}$ and write $z_q = f(p_{0q})$ and $e_q = |p_{mq} - z_q|$. A certified edge implies $z_i \geq z_j$ and $p_{mj} - p_{mi} \geq g_{ij}$. Thus

$$e_i + e_j \geq (z_i - p_{mi}) + (p_{mj} - z_j) \geq g_{ij}.$$

At least one endpoint has error at least $g_{ij}/2$. Summing over a matching counts each error at most once, so $\sum_q e_q \geq \sum_{\mathcal{M}} g_{ij}$. Both conclusions hold for every f, giving the stated infima. □

The matching bound is not generally the exact distance Δ_1. It is the largest bound of this disjoint-pair form. Summing overlapping pairs is invalid. Selecting the maximum edge or matching requires no new multiplicity penalty because all these deterministic implications hold on the original simultaneous event. The IID and chunk constructions retain separate error families. Selecting the larger of their answers therefore has no joint 95% guarantee. Zero bounds establish neither equivalence nor small error. The calculation retains all primary checkpoints and windows for descriptive comparison, but positive-bound summaries exclude outcome-selected runs. It quantizes edge weights downward at 10^{-12} probability resolution for integer-weight matching, and rounds displayed bounds downward to two decimal places.

D.8.3 Checking every finite-pool budget

The original grid-based analysis uses 12 recorded budgets. A separate calculation evaluates all integers $1 \le k \le 2048$ for all 117 full-budget primary checkpoints, and Figure 25 plots the 99 of them that carry a calibrated intervention. It averages the without-replacement failure probability over $B_q \sim \mathrm{Binom}(c_{0q}, a_m)$ retained parent successes. For $s_k(b) = \binom{2048-b}{k}/\binom{2048}{k}$, the stable recurrence is

$$s_0(b) = 1, \qquad s_k(b) = s_{k-1}(b) \max\left\{0, \frac{2048 - b - k + 1}{2048 - k + 1}\right\}.$$

This recurrence evaluates the finite-pool reference in Eq. 9, not the IID plug-in approximation. Of 86 nondegenerate comparisons, 12 have maximum absolute deviation at most one point. No checkpoint crosses that threshold when the grid is filled in. The other 29 all-zero children remain uninformative for reference shape, 16 of them inside the displayed subset. Qwen expert-75%'s largest deviation occurs at $k = 6$, outside the original grid. Neither the full-budget enumeration nor a small observed residual establishes population curve equivalence.

D.8.4 Confidence family and sampling assumptions

For a count c out of n IID Bernoulli draws, a two-sided Clopper–Pearson interval at error η uses the $\eta/2$ quantile of $\mathrm{Beta}(c, n - c + 1)$ as its lower endpoint when $c > 0$, and the $1 - \eta/2$ quantile of $\mathrm{Beta}(c + 1, n - c)$ as its upper endpoint when $c < n$. The other endpoints are zero and one, respectively. We assign $\eta = 0.05/11700$ to each interval slot in the complete recorded primary AIME grid, counting a shared parent once. The family reserves the full 2,048, prefix 256 and remainder 1,792 windows for every model and problem, including the six screened-out models whose larger windows were never observed. Those unobserved intervals are not computed and yield no certificate, but their correction slots are not removed.

For a prespecified run per checkpoint under the within-stream IID model, each window defines a potential fixed-budget sample even when continuation is unobserved. A union bound over the complete family gives simultaneous coverage with probability at least 95%. Screening only selects which intervals are reported, so the unconditional probability of any false reported certificate is at most 5%. The bound also allows scanning all problem pairs without an additional pairwise correction because every certificate is a logical consequence of that same simultaneous-coverage event. Independence between problems, models, or overlapping windows is unnecessary for the union bound. Within-stream IID sampling with unchanged probability is necessary for the binomial intervals. Neither fixed random seeds nor agreement across blocks establishes that assumption. This is not conditional coverage given selection, and the family does not cover retrospective adoption among completed runs, unrecorded model searches, outcome-chosen window boundaries, or arbitrary additional budgets.

The tolerance grid $\delta \in \{1, 2, 5\}$ is an exploratory sensitivity analysis, not a revision of the one-point discovery criterion discussed in Appendix E.5. Zero-retention cases remain in the All denominator but cannot be counted as informative near-reference fits. The non-FFN examples require a wider tolerance. Qwen expert-75% illustrates endpoint agreement only, with $D(2048) = 1.49$ points but $D(8) = -14.45$ points, and is therefore absent from these near-grid subsets.

Among the 114 inference-eligible primary checkpoints, 20 have at least one identical certified pair in all three windows. This overlap is a stability description, not a second confidence level. Certificates concern the fixed grader and generation policy. They do not validate the underlying reasoning or isolate an architectural cause.

Appendix D.9.2 derives what a complete curve identifies and gives the complementary example showing why aggregate agreement cannot recover problem identities.

D.8.5 ALLOWING DEPENDENCE WITHIN GENERATION CHUNKS

The binomial intervals above assume independent candidate successes within a model–problem stream. The server returns 32 candidates per request, so we also treat a whole request as one observation. Let $X_b \in [0,1]$ be the fraction of correct candidates in chunk b. We permit arbitrary dependence within a chunk and assume only $\mathbb{E}[X_b \mid \mathcal{F}_{b-1}] = \mu$ for the ordered chunks of a stream. Independent chunks with a common mean suffice. Common marginal means alone do not suffice, and distinct random-seed labels do not establish this conditional-mean property. Cross-problem, cross-model and overlapping-window dependence is still allowed.

We use fixed mixtures of nonnegative betting products (Waudby-Smith & Ramdas, 2024). For the six fractions $\mathcal{C} = \{0.1, 0.25, 0.5, 0.75, 0.9, 1\}$, define

$$E_R^+(\mu) = \frac{1}{6}\sum_{c\in\mathcal{C}}\prod_{b=1}^{R}\left(1 - c + \frac{cX_b}{\mu}\right), \qquad E_R^-(\mu) = E_R^+(1-\mu; 1-X_1, \ldots, 1-X_R). \tag{14}$$

Each factor is nonnegative and has conditional expectation one at the true mean. Iterated expectation therefore gives $\mathbb{E}[E_R^{\pm}(\mu)] = 1$. Markov's inequality controls each tail, and monotonicity allows conservative bisection inversion into an interval. The lower endpoint is zero for an all-zero stream, and the upper is one for an all-one stream. Zero factors in the $c = 1$ product are retained, not smoothed. The fixed mixture averages capital, not interval endpoints, and does not select a betting fraction from the data.

The analysis explored three interval constructions, namely a Hoeffding bound, a Bernoulli-KL Chernoff bound for bounded means, and the fixed betting mixture. To account for choosing among these three specified methods, we allocate each method $0.05/3$ across the same $M = 11520$ reserved model–problem–window slots. Each tail uses threshold $6M/0.05$ in Eq. 14. For prespecified runs, a union bound then controls false certificates over this three-method sensitivity family by 5%. The finite-family correction covers method choice, not retrospective run adoption. The original binomial family is reported separately. No 5% joint guarantee for selecting between binomial and chunk-level certificates is claimed. The two simpler bounds produce no full-pool certificates in this cohort.

The full, prefix and remainder windows contain 64, 8 and 56 chunks. Eight chunks cannot resolve any ordering at this correction level. At $\mu = 1/2$, each factor is at most two, so $E_8^{\pm}(1/2) \le 2^8$ is below the rejection threshold. Every prefix interval therefore contains $1/2$, and strict prefix certificates and three-window intersections are structurally impossible. Their zeros are not evidence of failed replication. Full–remainder agreement is reported only descriptively because those windows overlap.

These intervals concern average success probabilities. They do not turn within-chunk dependent sampling into IID population pass@k, identify the historical served weights, or remove the need for independent new generations. Finite-pool coverage remains a conditional uniformly sampled-subset statistic and requires no IID approximation.

Independent-seed replication remains unavailable. Its proposed design and protocol limitations are separated in Appendix E.5.

D.9 WHAT INFORMATION DO THE CURVES IDENTIFY?

This subsection supplies the mathematical basis for the reference and ordering analyses in Appendices D.1 and D.2, followed by a secondary curve-fitting diagnostic. Finite-pool concavity and factorial moments explain coverage changes. Inversion identifies the count histogram but not its assignment to problems. Neither these identities nor fitted coefficients identify a latent capability decomposition.

D.9.1 FINITE-POOL ALLOCATION AND FACTORIAL MOMENTS

The following elementary consequences of discrete concavity and inclusion–exclusion explain what the diagnostic measures. They are not claims of new underlying mathematical inequalities.

Proposition 3 (Finite-pool allocation ordering). *Let $Q = |\mathcal{Q}|$, and let every problem have n candidates and equal weight, with $P_{\boldsymbol{c}}(k) = Q^{-1}\sum_q g_k(c_q)$ and $g_k(c) = 1 - \binom{n-c}{k}/\binom{n}{k}$. For integer vectors $\boldsymbol{x}, \boldsymbol{y} \in \{0, \dots, n\}^Q$ of equal sum, suppose $\boldsymbol{x}$ majorizes $\boldsymbol{y}$, meaning that their decreasing rearrangements obey $\sum_{q=1}^{r} x_q^{\downarrow} \geq \sum_{q=1}^{r} y_q^{\downarrow}$ for every $r < Q$. Then $P_{\boldsymbol{y}}(k) \geq P_{\boldsymbol{x}}(k)$ for every $1 \leq k \leq n$, with equality at $k = 1$.*

Proof. Writing $g_k(c) = 1 - \binom{n-c}{k}/\binom{n}{k}$ and taking a binomial coefficient as zero when its lower argument exceeds its nonnegative upper argument, Pascal's identity gives

$$g_k(c+1) - g_k(c) = \frac{\binom{n-c-1}{k-1}}{\binom{n}{k}}, \qquad 0 \leq c < n.$$

These increments are nonincreasing in c. Moving one success from count r to count s, where $r \geq s + 2$, changes the summed coverage by $[g_k(s+1) - g_k(s)] - [g_k(r) - g_k(r-1)] \geq 0$. Integer majorization is generated by these balancing transfers and permutations. For $k = 1$, every increment is $1/n$, establishing equality. □

The statement is sufficient and weak, not a converse characterization of coverage curves. For example, with $n = 5$, neither $(4, 4, 0)$ nor $(5, 2, 1)$ majorizes the other, yet the latter has at least as much coverage at every supported budget. Unequal pool sizes or problem weights require a different ordering statement.

Proposition 4 (Factorial moments under thinning). *Define $M_{t,j} = Q^{-1}\sum_q (c_{tq})_j/(n_{tq})_j$ for supported j, where $(z)_j = z(z-1)\cdots(z-j+1)$. For budgets supported by both models,*

$$P_t(k) = \sum_{j=1}^{k} (-1)^{j+1} \binom{k}{j} M_{t,j}, \tag{15}$$

$$D_m(k) = \sum_{j=1}^{k} (-1)^{j+1} \binom{k}{j} \left(M_{m,j} - a_m^j M_{0,j}\right). \tag{16}$$

In particular, without assuming matched means on the evaluation block,

$$D_m(2) = 2(M_{m,1} - a_m M_{0,1}) - (M_{m,2} - a_m^2 M_{0,2}).$$

Proof. For a uniformly selected k-candidate subset, any specified j positions are all successful with probability $(c)_j/(n)_j$. Inclusion–exclusion for at least one success yields the first identity. For $B \sim \mathrm{Binom}(c, a_m)$, each ordered j-tuple of parent successes survives with probability a_m^j, so $\mathbb{E}[(B)_j] = a_m^j (c)_j$. Substitution gives the second identity and its $k = 2$ specialization. □

The moment identity permits different pool sizes because each denominator is normalized separately. In-sample mean matching cancels its first-order term, whereas candidate-block transport need not. Moment scaling holds in expectation over thinning, not for every realized retained pool. Because alternating sums at large k are numerically ill-conditioned, the numerical calculation uses the equivalent hypergeometric expectation.

D.9.2 Curve information and missing problem identities

Exactly what a complete curve identifies. When n is common and the entire unrounded finite-pool curve is available, binomial inversion recovers

$$M_j = \sum_{k=1}^{j} (-1)^{k+1} \binom{j}{k} P(k), \qquad H(z) = \frac{1}{Q}\sum_q z^{c_q} = \sum_{j=0}^{n} \binom{n}{j} M_j (z-1)^j,$$

where $M_0 = 1$. The coefficients of H uniquely determine the normalized success-count histogram and, given Q, the count multiset. They do not recover problem labels. This exact identification result is not an instruction to invert rounded, sparsely sampled, or noisy high-budget curves numerically.

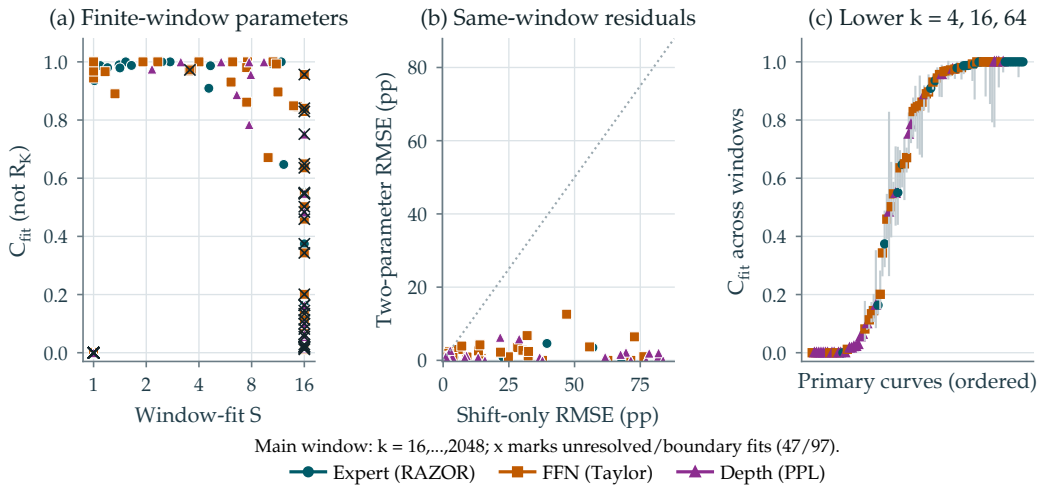


Figure 28: Finite-window alignment diagnostics. Pure horizontal alignment and horizontal-plus-vertical alignment are compared on an identical observed window. Expert, FFN and calibrated depth pruning are shown, and tail-drop is omitted. Fits at the upper shift bound or with weakly identified shifts remain marked. Window sensitivity limits interpretation of the fitted direction. Neither the vertical coefficient nor a residual is an estimate of mathematical support.

Why the curve alone cannot answer this question. For population probabilities $\boldsymbol{p}_0 = (0.8, 0.2)$ and $\boldsymbol{p}_m = (0.2, 0.8)$, IID population coverage is identical for every k, with $\Pi_0(k) = \Pi_m(k) = 1 - (0.8^k + 0.2^k)/2$. Nevertheless, the problem order is reversed. Permuting equal-size finite-pool counts likewise leaves Eq. 1 unchanged. Therefore aggregate agreement, even at every budget, cannot certify preservation of problem identity. The moment and rank diagnostics are complementary. The former describes success allocation without labels, while the latter tests its alignment with the parent problems.

D.9.3 Curve alignment and its identifiable range

As a secondary descriptive check, we fit

$$P_m(k) \approx C_m^{\text{fit}} P_0(k/S_m^{\text{fit}}) + \epsilon_m(k). \tag{17}$$

The vertical coefficient differs from endpoint retention because

$$R_{m,K} \approx C_m^{\text{fit}} \frac{P_0(K/S_m^{\text{fit}})}{P_0(K)} + \frac{\epsilon_m(K)}{P_0(K)}. \tag{18}$$

Thus even a pure horizontal shift can reduce the observed endpoint at fixed K. The main window is $k = 16, 32, \ldots, 2048$ with $1 \le S \le 16$ and $0 \le C \le 1$, using interpolation in log budget. Shifts at the upper bound and flat profiles are flagged. In the 108-checkpoint full-budget grid excluding Gemma, 63 fits receive a zero-curve, upper-shift, or flat-profile flag. These diagnostics do not identify a latent capability decomposition and are not the primary rate-controlled reference of the paper.

The fitting diagnostics are computed on the full-budget primary cohort excluding Gemma, which keeps the tail-drop screens in that reference count. Figure 28 instead shows the 99 calibrated curves with Gemma included, so the two cohorts differ by design rather than by omission. For a fixed shift $s = \log_2 S$, let $a_j = P_0(2^{u_j - s})$ and $b_j = P_m(2^{u_j})$ on the selected log-budget grid. The constrained least-squares vertical coefficient has the analytic solution

$$\widehat{C}(s) = \text{clip}_{[0,1]} \left(\frac{\sum_j a_j b_j}{\sum_j a_j^2} \right), \tag{19}$$

when the denominator is positive. We profile the residual over s and compare it with the residual under $C = 1$. If the parent curve is zero on the window, the alignment is not identifiable. If the pruned curve is zero, the best zero scale does not identify a horizontal shift.

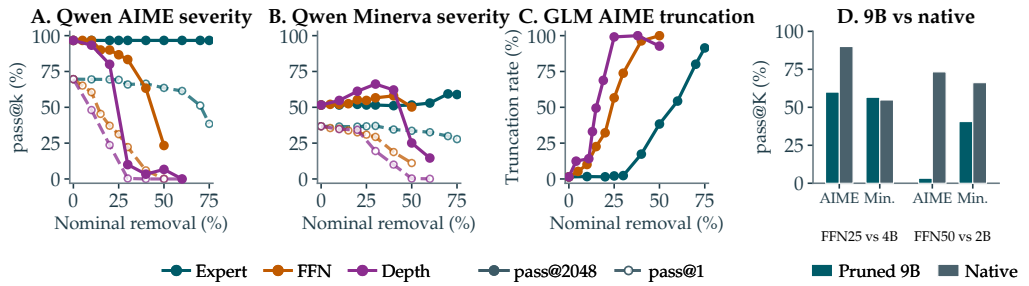


Figure 29: **A joint view of pruning outcomes.** Qwen pass@1 and coverage by pruning axis on AIME and Minerva, GLM AIME truncation, and pruned Qwen3.5-9B versus native 4B/2B. Full profiles appear in Appendices B.1, B.7 and B.5.3.

The main window begins at 16 samples and allows at most a 16-fold shift. Beginning at 4 or 64 changes both the retained budgets and the largest allowed shift. No candidate shift changes the mask of fitting observations within a window, so every shift is evaluated on the same observations. The parent is interpolated only inside its observed domain. Values below one sample are neither clipped to the parent's first point nor extrapolated through an arbitrary probability model.

A low fitted C may reflect vertical deformation, an insufficient allowed shift, or a limited parent curve. A fitted C near one likewise does not prove unchanged capability. We therefore interpret the fit through its residual profile, boundary status, and sensitivity to the window. The empirical $S(y)$ and fixed-budget R_K do not rely on this fitting assumption.

E Extended Discussion and Open Questions

This appendix develops implications for pruning choices and the limits of accepted-answer measurement. A bounded knowledge-task pilot and a secondary offline-allocation diagnostic retain their limited scope and negative results. The allocation analysis reuses recorded pools and is not a proposed pruning or deployment method. We conclude with the limits of independent-seed replication and the validation that remains unavailable. Available trajectory measurements and decoding controls are reported in Appendices C and D.5.

E.1 What the joint profile implies for pruning choices

Pruning changes success frequency, sampled solvable range and generation behavior in different ways. Coverage retention or expansion can coexist with lower single-sample accuracy, but need not require it. The contrasting backbone and axis profiles preclude a universal preservation rule. Selection criteria can differ in both coverage and sampling efficiency, including substantial matched-budget differences between calibrated criteria on Gemma. Model-origin comparisons ask a separate question and remain conditional on the task, native release and dimensions matched.

Implications for choosing a pruning configuration. Evaluate the candidate's full sampling profile and generation behavior on each intended task, rather than transferring a threshold from another backbone or benchmark. Compare selection methods within a matched pruning axis and report the actual dimensions matched in native-small comparisons. A configuration can retain a broad solvable range yet require many more attempts, and an apparently improved endpoint can replace some parent-solved problems. These are reasons to report a joint profile, not to collapse the findings into a single universal pruning score.

E.2 What does accepted-answer coverage establish?

A large sampling budget amplifies grader weaknesses as well as rare successes. Minerva includes short numerical answers that can be matched accidentally by degenerate outputs. It also includes equivalent numerical forms that a strict symbolic parser may treat differently. These effects can

operate in opposite directions and need not affect parent and pruned models equally. We retain the common strict grader for the main comparisons and do not promote Minerva's large-budget endpoint to a capability boundary.

The corrected grading also leaves the Minerva denominators uneven. Gemma coverage is computed on all 272 problems, while the other backbones remain on the 219 retained problems pending the same correction (Appendix A.4). The 53 restored problems are not a random sample, because they are exactly the ones whose reference answers the original parser could not read, so their difficulty need not match the retained set. Absolute Minerva coverage is therefore not comparable across backbones, and backbone comparisons should be read as statements about parent-relative change rather than about absolute levels. Comparisons within a backbone, including every parent-child contrast and budget sweep, use one fixed problem set and are unaffected. AIME-2026 uses all 30 problems for every backbone and is free of this asymmetry.

AIME avoids some numerical-format ambiguities but is not immune to guessing. Its integer-answer format alone cannot justify that claim. For example, uniform independent guesses over 1,000 possible answers would hit the correct answer with probability $1 - (999/1000)^{2048}$, approximately 87%. This calculation is not a model of the actual generated distribution. It shows why low single-sample chance accuracy does not automatically make large-budget coverage evidence of valid reasoning. Accepted-output inspection, an appropriate verifier, and sensitivity to the answer distribution remain important.

The primary numerical claims concern observed grader-based candidate coverage. They are strongest when differences recur across budgets and in executed-code evaluations, rather than resting on one newly solved short-answer problem. No endpoint observation here identifies the set of all outputs that a model could generate with unlimited sampling or a different decoder.

Appendices C.4 and C.5 complement these coverage measurements with trajectory statistics and identify the annotations needed to assess reasoning validity and valid-strategy diversity. Appendix D.5 separates available decoding controls from the sensitivity checks that remain incomplete.

E.3 A BOUNDED KNOWLEDGE-TASK PILOT

A separate exploratory cohort evaluates four parents and ten pruned configurations on 24 questions from each of MMLU-Pro (Wang et al., 2024) and GPQA-Diamond (Rein et al., 2024), with 4 candidates per question. This is a small-subset pilot, not a full-benchmark evaluation or an extension of the primary $n = 2048$ grid. MMLU-Pro contributes six questions each from biology, chemistry, computer science and physics. GPQA contributes 24 sampled Diamond questions. Selection uses seed 20260911 and identical questions and ordering for all models. The GPQA questions overlap with an earlier parent-only evaluation and are therefore not previously unseen held-out items.

Generation uses one four-candidate request per question, base seed 42, temperature 0.7, top-p 0.95, top-k 20, repetition penalty 1.0 and an 8,192-token cap. The requested chat-template setting disables dedicated thinking where supported, but the prompt still requests step-by-step reasoning and a boxed option letter. The grader extracts the option letter from the last boxed field and compares it with the reference letter. Missing or invalid letters score zero.

All 2688 saved responses were checked by extracting the last boxed option letter and comparing it with the reference. The resulting labels were compared with recorded scores. Per-problem counts and aggregate $\mathrm{pass@1}$, $\mathrm{pass@2}$ and $\mathrm{pass@4}$ were recomputed from these labels. These checks concern answer extraction and score aggregation, not reasoning validity or the identity of all historically served weights. Undefined aggregate standard errors are excluded rather than interpreted as zero uncertainty.

Let $A_{tq} = \mathbf{1}\{c_{tq} > 0\}$. With L parent-only and G child-only successes, the exact endpoint identity is $P_m(4) - P_0(4) = (G - L)/24$ in fractional units. Equal coverage can therefore conceal an exchange of solved questions. Qwen FFN-20% and its parent each cover 19/24 GPQA questions, but lose 1 and gain 1. GLM expert-25% and its parent each cover 21/24 MMLU-Pro questions, also with one lost and one gained. These are observed set differences, not probability-order certificates, since four candidates do not establish the underlying success ordering. The pilot retains all 20 parent–child/task contrasts, including unchanged sets and counterexamples to a universal axis ranking.

Low scores also accompany response failures. Across the six task results for 25% depth pruning of GLM, Qwen3.5 and Llama, 85.4–96.9% of outputs lack a valid boxed letter, while 28.1–95.8% hit the token cap. These are overlapping, not identical, failure sets. A gradeable gold answer and an error-free API request do not imply a parseable model answer. The scores consequently cannot isolate erasure of parametric knowledge from failure to produce the required response.

Multiple-choice guessing remains a distinct limitation. With M equally likely choices, independent random answers cover a correct option with probability $1-(1-1/M)^K$. At $K=4$, the four-choice GPQA reference is already 68.36%. This is a hypothetical guessing reference, not a performance floor for any model. Parent normalization does not cancel saturation. This pilot supports finite-pool measurement distinctions, not knowledge improvement, full-benchmark generalization or cross-seed replication.

E.4 Can changed success frequencies inform candidate allocation?

Changed success frequencies raise a secondary question. Does child recalibration alter coverage under a fixed allocation rule? The following offline diagnostic reuses disjoint candidate blocks on the same problems, retaining calibration costs, negative results and degenerate cases. It tests planning transfer, not a deployment method or unseen-task generalization.

E.4.1 Transfer to disjoint candidate blocks

For each primary AIME checkpoint, we estimate probabilities from the first 256 candidates and evaluate allocations on the disjoint remaining 1,792. The reverse direction uses 1,792 for estimation and 256 for evaluation. Every policy receives the same test budget Qb, with $b \in \{16, 32, 64, 128\}$ and a per-problem cap equal to the test-pool size. The test counts never enter policy construction.

We compare four static policies. Uniform allocation assigns b candidates to each problem. The parent policy uses $\widetilde{p}_{0q} = (c_{0q} + 1/2)/(n_{\text{train}} + 1)$. The mean-adjusted parent policy uses $\widetilde{a}\widetilde{p}_{0q}$, where $\widetilde{a} = \overline{\widetilde{p}}_m/\overline{\widetilde{p}}_0$ is estimated from the same training block. Values $\widetilde{a} > 1$ are outside this loss-only policy and remain NA. The child-recalibrated policy uses $\widetilde{p}_{mq}$ directly. The fixed half-count prior is a planning heuristic, not smoothing of the observed coverage estimator. This smoothed-mean ratio differs from the raw-mean thinning rate in Eq. 9.

For a policy probability vector $\boldsymbol{v}$, integer allocations maximize

$$\sum_q \left[1-(1-v_q)^{k_q}\right] \quad \text{subject to} \quad \sum_q k_q = Qb, \qquad 0 \le k_q \le n_{\text{test}}.$$

Each additional sample has marginal value $v_q(1-v_q)^{k_q}$, which decreases with k_q. Repeatedly choosing the largest available marginal therefore solves this separable discrete-concave surrogate exactly. Ties use a fixed natural problem-ID order. The surrogate plans samples but does not estimate performance. Evaluation instead computes

$$\frac{1}{Q}\sum_q \left[1 - \frac{\binom{n_{\text{test}}-c_{mq}^{\text{test}}}{k_q}}{\binom{n_{\text{test}}}{k_q}}\right],$$

with zero assigned to $k_q = 0$. This is conditional expected coverage under uniform subsampling of each recorded test pool. No candidate generation is rerun.

Costs and inferential scope. All policies are charged Qn_{train} child-calibration candidates plus Qb test candidates, and parent-informed policies also use previously collected parent calibration samples. Equalizing these experimental charges does not equalize necessary deployment costs, because uniform allocation needs no labels, and parent-only planning needs no child labels. Next-batch coverage excludes calibration successes, so it cannot establish total compute savings over an oracle that stops at the first success.

The main contrast is child recalibration minus mean-adjusted parent planning. It reflects probability-shape changes and estimation error together. A gap is not a causal estimate of the effect of rank reversals. The full analysis retains all 117 checkpoints, both directions and every stated budget,

including cases where recalibration does worse. For four illustrative forward $b = 64$ cases, 1000 paired-problem bootstrap replicates reestimate both probability vectors and $\widetilde{a}$ and rerun the allocation. We report the central 95% range of admissible resamples and their count. If some have $\widetilde{a} > 1$, discarding those undefined comparisons does not establish conditional or unconditional 95% confidence coverage. These ranges describe problem resampling of recorded pools, not fresh-generation uncertainty or a calibrated null-rejection test.

E.4.2 Matched-marginal assignment and cumulative coverage

Two controls distinguish the probability multiset from its assignment to problem identities, and next-batch from cumulative coverage. Retaining calibration successes under unchanged allocations makes the child-minus- mean-adjusted-parent gain nonpositive in all four examples at this budget, despite some positive next-batch gains. This is not universal, because the calibration-depth sweep also contains positive cumulative gains. Appendix E.4.3 then tests whether skipping solved problems yields a difficulty-order advantage, rather than merely using the solved-set mask.

The mean-adjusted parent policy leaves changes in the entire probability multiset uncontrolled. We add a stronger descriptive comparison. Sort the parent's training estimates in decreasing order, breaking equal-probability ties by increasing natural problem ID. Assign the child's sorted training probabilities, also decreasing, to those positions and rerun the same greedy allocator. This *parent-order control* preserves the complete child probability multiset. Consequently, it preserves every uniform-budget surrogate score and the optimal separable planning objective under a common total budget and per-item cap because permuting the probability vector permutes a symmetric feasible allocation set. Held-out coverage can nevertheless differ because the true problem identities have not been permuted.

This control uses child as well as parent calibration scores, not a parent-only transfer method. Its contrast includes estimation error and arbitrary resolution of parent ties, not a causal effect of certified rank reversals. A second fixed rule reverses IDs within parent ties while retaining all strict parent orderings. Both rules are computed from training data only and reported without selecting the better held-out result. With 256 calibration candidates, Qwen FFN-5%'s next-batch difference changes from -0.11 to $+0.16$ points across these rules. Neither direction supports a robust planning advantage. The ranges across two rules are sensitivity ranges, not confidence intervals.

At $t = 256, b = 64$, the complete 117-checkpoint cohort has 66 positive, 8 negative and 41 numerical-zero next-batch child-minus-control differences under increasing-ID ties, versus 67/7/41 under decreasing-ID ties. Retaining calibration successes changes those counts to 9/37/69 and 15/35/65, respectively, in the same positive/negative/zero order. There are 38 all-zero child calibration blocks, which contain no child probability-order information. The analysis includes all checkpoints, budget settings and both tie rules. The four examples in the tables illustrate these comparisons but do not estimate representative frequencies or supply independent replications.

Changing the outcome without changing the allocation. Let $s_q = \mathbf{1}\{c_{mq}^{\mathrm{cal}} > 0\}$ indicate an actual success in the specified calibration pool, and let $h_q(k)$ be the held-out finite-pool coverage of k candidates. For a fixed plan A with allocation k_q^A, define

$$C_{\mathrm{next}}(A) = \frac{1}{Q}\sum_q h_q(k_q^A), \qquad C_{\mathrm{incl}}(A) = \frac{1}{Q}\sum_q \left[s_q + (1-s_q)h_q(k_q^A)\right]. \tag{20}$$

The specified calibration candidates are all retained, not randomly subsampled from a pooled calibration-plus-test set. The next allocation is unchanged, including samples assigned to already-covered problems. Thus this is an outcome-accounting sensitivity, not a solved-aware replanning or early-stopping algorithm. For two policies A, B, exact subtraction gives

$$\Delta_{\mathrm{incl}}(A,B) = \frac{1}{Q}\sum_q (1-s_q)\left[h_q(k_q^A) - h_q(k_q^B)\right]. \tag{21}$$

The difference $\Delta_{\mathrm{next}} - \Delta_{\mathrm{incl}}$ is precisely the next-batch gain on problems already covered in calibration. Each policy's inclusive coverage is at least its own next-batch score, yet a between-policy gain can reverse sign. That reversal occurs without changing a single allocation. Per-problem contributions and both parent-tie rules are retained for every checkpoint.

Calibration depth and budget accounting. We vary both models' calibration prefixes over $t \in \{32, 64, 128, 256\}$ candidates, using one, two, four or eight complete requests. The held-out pool always consists of chunks 8 through 63, with 1,792 candidates. Intermediate chunks are unused when $t < 256$. One design fixes the next-batch mean budget at $b = 64$. A second fixes the total child budget at $t + b = 320$. The latter trades off calibration and subsequent sampling, rather than isolating estimation quality. The former still changes both models' estimates and the calibration-covered set. Both are joint calibration-depth sensitivities, not independent repeats. The original reverse direction remains a separate window-stability comparison.

The conclusion depends on the operating point rather than ruling out any benefit from recalibration. For GLM expert-50% at $t = 64$, child-informed inclusive coverage is 88.92% versus the mean-adjusted parent's 87.60%. At $t = 256$, those scores are 81.49% and 83.73%, respectively, under the same total child budget. More calibration both changes the estimates and leaves fewer candidates for the next batch. This comparison does not isolate a causal calibration effect or select a validated optimal budget.

At each depth, the accounting is $Qt + Qb$ child candidates, with Qt previously collected parent candidates counted separately. Labels come from automatic grading, not human annotations, and both outcomes measure oracle coverage, not answer-selection accuracy. Reallocating recorded pools does not change their historical generation cost. All 117 checkpoints are retained, including zero-success blocks and $a > 1$ cases. The latter invalidate only the scaled-parent comparison, not the parent-order control. Sign counts use a 10^{-9}-point numerical-zero tolerance and are descriptive, not tests across independent models.

E.4.3 An exact count-only degeneracy when skipping solved problems

The preceding inclusive comparison changes only the outcome accounting. The next comparison changes the allocation while retaining the same fixed calibration and test windows. Let $\mathcal{U} = \{q : c_{mq}^{\mathrm{cal}} = 0\}$ be the problems not yet solved in child calibration. With integer $0 \le b \le N$, a next-batch allowance $B = Qb$ and common held-out cap $N = n_{\mathrm{test}}$, use

$$B_{\mathrm{used}} = \min\{B, |\mathcal{U}|N\}, \qquad k_q = 0 \ (q \notin \mathcal{U}), \qquad \sum_{q \in \mathcal{U}} k_q = B_{\mathrm{used}}. \tag{22}$$

Any unused allowance is reported, not spent on already-solved problems. For each original probability-based policy, freeze its fitted probabilities and rerun the greedy allocator on $\mathcal{U}$ only. The uniform policy divides B_{used} equally over this subset, assigning any remainder in increasing natural-ID order. An empty subset receives no samples. All allocations are fixed before held-out scoring, without within-batch feedback or online stopping. Coverage remains C_{incl} with denominator Q, not $|\mathcal{U}|$.

Proposition 5 (Count-only planning after a zero-success filter)**.** *With equal calibration depth t, the estimate $\widehat{p}_{mq} = (c_{mq}^{\mathrm{cal}} + 1/2)/(t + 1)$, a common cap and the same natural-ID tie rule, the child-informed greedy allocation on $\mathcal{U}$ is exactly the uniform allocation on $\mathcal{U}$. Their allocation vectors, held-out coverage and cumulative coverage coincide for every allowed budget.*

Proof. Every $q \in \mathcal{U}$ has $\widehat{p}_{mq} = 1/[2(t + 1)] = p$. Its next marginal gain is $p(1 - p)^{k_q}$, strictly decreasing in k_q. Greedy therefore always chooses a currently least-sampled unsolved problem, with the specified ID rule resolving ties. Before the common cap binds, allocations differ by at most one and follow the same remainder rule as uniform allocation. At saturation all equal the cap. If $\mathcal{U}$ is empty, both allocate zero. Scoring identical vectors gives identical scores. □

This is a property of the specified count-only estimator and equal-depth protocol, not an impossibility result for richer child predictors, unequal sampling histories or all recalibration algorithms. Matching the child's probability multiset *within* $\mathcal{U}$ would also give this constant vector, whatever parent order is used. Conversely, masking the earlier full-set parent-order control need not preserve the multiset on $\mathcal{U}$, so its matched-marginal interpretation does not extend to this comparison.

All four policies share the child-derived solved mask. The masked parent policy is therefore not a child-label-free deployment method. The adjusted parent retains the original full-Q scale a without refitting it to match the mean on $\mathcal{U}$, and $a > 1$ remains NA. Total child usage is $Qt + B_{\mathrm{used}}$ rather than automatically $Qt + B$, with previously collected parent calibration samples accounted for separately. These are counterfactual candidate counts, not measured generation savings or verifier costs.

For each policy separately, this change cannot lower its old inclusive score. Holding probabilities, caps and tie rules fixed, greedy orders a fixed list of marginal gains. Restricting that list to unsolved problems cannot remove an already-selected unsolved marginal from the first B choices. If the restricted list is shorter, it is exhausted. Thus every unsolved problem receives at least its old allocation. The same inequality holds for uniform allocation since $B_{\text{used}} \geq |\mathcal{U}|b$. Finite-pool coverage is nondecreasing in allocation, proving the claim. This per-policy dominance does not determine the sign of a between-policy gap. With a single unsolved problem or enough allowance to exhaust every unsolved pool, all valid policies necessarily agree. These cases must be distinguished from an empirical absence of ranking benefit.

Full-cohort results. The same four examples carry over here, without selecting new cases. At $t = 256, b = 64$, skipping solved problems raises GLM expert-50%'s child-planned cumulative coverage from 81.49% to 85.75%, but the adjusted-parent plan still reaches 86.25%. The within-policy improvement comes without a child-ranking advantage because C and U are exactly the same plan. Across all 117 checkpoints at this setting, C minus P is positive/zero/negative in 53/46/16 cases, while C minus M gives 31/46/38. No calibration set is fully solved. One checkpoint has a single unsolved problem whose 1,792-candidate pool is exhausted, leaving 128 of the 1,920-candidate next allowance unused. All other checkpoints use the full allowance. Rounded score agreement alone does not imply equal plans for P or M.

The full cohort covers all 117 checkpoints under each of the eight settings. The reverse window has 40 fully saturated unsolved sets, including eleven with only one unsolved problem, versus one at the main $t = 256, b = 64$ setting. Its many zero gaps therefore reflect a different operating regime, not merely a lack of ranking benefit. The exact C=U equality (Proposition 5) and per-policy cumulative dominance do not imply a positive child-minus-parent gap. These are same-pool diagnostics, not independent replications or calibrated tests.

E.5 Limits of independent-seed replication

The exploratory near-reference FFN-5% result lacks independent-seed replication. A proposed two-seed check would compare the same parent and child on all 30 AIME problems with 2,048 candidates per problem, retaining the five discovery-certified directed pairs and the original 12-budget grid. Its criterion is at least one certified fixed pair per seed and a maximum absolute coverage difference of one percentage point from the reference across the grid. This is neither the all-integer-budget check (Appendix D.8.3) nor a population-equivalence test. The proposed Bonferroni family contains $2 \times 2 \times 30 = 120$ two-sided binomial intervals under the same IID assumption. This check remains unperformed because the historical pruned checkpoint is inaccessible. The parent's historical repetition penalty is also unknown, so setting it to 1.0 would not establish exact protocol replication.

E.6 Remaining validation needs

Validation still requires mean-gradient re-pruning to test the FFN scorer limitation (Appendix A.3) without recovery training, representative derivation audits to estimate invalid-reasoning prevalence, and a complete hardware latency/throughput grid. These results remain unavailable. The present comparisons establish neither a causal decomposition nor a hardware ranking.