# BUDGETS TRANSFER, HARDNESS LABELS DO NOT: AUDITING WHAT SURVIVES QUANTIZATION OF REASONING MODELS

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

## ABSTRACT

Reasoning models are usually quantized just before deployment, but two settings that guide their use are measured earlier, on the full-precision (BF16) model: the decode budget, which caps how many tokens the model may generate, and prompt hardness labels, which decide which prompts receive extra training. We ask whether these BF16 measurements can be reused after quantization. A direct comparison is misleading, because a longer budget can look better only because fewer answers are cut off, and because two runs of the same model already disagree on which prompts are hard. We therefore check truncation before comparing budgets, and compare every cross-precision change in the hard set with an independent rerun of the same model. On Qwen3.5-4B with 4-bit GPTQ, the two settings behave differently. The budget can be reused: serving the 4-bit model at the budget chosen on BF16 changes GPQA accuracy by $-0.59$ points on average over six seeds ($95\%$ interval $[-1.94, +0.76]$). Hardness labels cannot. The BF16 and 4-bit hard sets overlap by $0.414$ (Jaccard), well below the $0.593$ between two BF16 runs. The lower overall accuracy of the 4-bit model does not explain the gap: shifting and rescaling the BF16 scores to match it still predicts an overlap of $0.523$. Quantization changes *which* prompts are hard, not only how many. A second quantizer, round-to-nearest, shows the same mismatch. The practical rule is to inherit the budget from BF16 but remeasure hardness on the deployed model, and to test any hardness-based training against a random control, since in our runs it gives no detectable gain.

## 1 INTRODUCTION

Reasoning models are commonly served under explicit token budgets, and weight-only quantization is widely used to reduce their serving cost (Frantar et al., 2023; Lin et al., 2024). Quantization typically comes last in the pipeline, after two decisions have already been made on the BF16 model. The decode budget is chosen by sweeping candidate caps on an evaluation set (Wu et al., 2025; Muennighoff et al., 2025), and prompt hardness is labeled by sampling several traces per prompt and counting how many are correct, to decide which prompts receive more training (Lu et al., 2026; Zeng et al., 2025). Both measurements are expensive, so they are made once and reused for every quantized variant that is later served. There are concrete reasons to expect this reuse to fail. Quantized reasoners lose accuracy (Liu et al., 2025), can generate longer traces (Lian et al., 2026), and change their failure modes (Oladri et al., 2026), so the best budget could move and different prompts could become hard. A stale budget sets the compute and accuracy of every request, and a stale hardness label sends repeated training exposure to the wrong prompts. Prior work nonetheless evaluates quantized models at a fixed budget and does not ask whether the BF16 decisions still hold (§2). We ask this question directly: *which BF16 measurements can a quantized reasoning model inherit?*

A naive comparison across precisions can mislead for three reasons. **(C1)** A longer decode budget may look better only because it lets more truncated traces finish, not because it is a better serving choice. **(C2)** Hard-set membership is itself stochastic: two runs of the same configuration label different prompts as hard. **(C3)** Even movement beyond this sampling noise does not by itself mean that hardness must be remeasured. If quantization only shifted or rescaled prompt hardness globally, one transform could correct the BF16 scores. We do not propose a new quantizer or training method;



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.

instead, an audit protocol (§3) turns each confound into an explicit control. For C1, we decode each prompt from the same random stream at every candidate budget and name a best budget only when fewer than $5\%$ of traces are cut off at the longest one. For C2, we read every cross-precision hard-set overlap against an independent repeat of the same configuration. For C3, we fit the simplest global correction, a shift and rescaling of per-prompt hardness on the logit scale, and simulate it forward to see whether it reproduces the observed overlap. A final check compares hardness-based training with a random selection of the same size and repetition.

Applied to Qwen3.5-4B with GPTQ-W4, the audit shows a clear asymmetry. **The budget transfers.** Reusing the BF16 budget on W4 costs $-0.59$ pp on GPQA ($[-1.94, +0.76]$, §4.1). **Hardness does not.** The BF16–W4 hard-set Jaccard is $0.414$, against $0.593$ for an independent repeat (§4.2), and the fitted global correction still predicts $0.523$ (§5). Matching the observation requires a prompt-specific effect of about one logit under GPTQ, larger than any BF16 or W4 repeat. A second quantizer, RTN-W4, reproduces the mismatch. Hard sets also appear to move across budgets, but this movement disappears once truncated traces are removed. Finally, hardness-based SFT shows no detectable gain over a matched random selection.

Our contributions are (i) the transfer question and an audit protocol that separates real transfer failure from truncation, sampling noise and a global accuracy drop; (ii) evidence of an asymmetry on one fully audited case; and (iii) a practical rule: inherit the budget, remeasure hardness on the deployed model ($G{=}8$ mid-budget traces per prompt, Appendix A.11), and test any hardness-based allocation against a matched random control. The case study covers one model family, so we present it as a tested protocol with one complete application rather than a general law.

## 2 Related Work

Post-training quantization is typically evaluated using perplexity, task accuracy, memory, and throughput: GPTQ and AWQ are widely used weight-only methods (Frantar et al., 2023; Lin et al., 2024), LLM.int8() and SmoothQuant address activation outliers (Dettmers et al., 2022; Xiao et al., 2023), and OmniQuant and EfficientQAT extend quantization to more difficult low-bit settings (Shao et al., 2024; Chen et al., 2025). Standard evaluation can miss changes in multi-step reasoning (Liu et al., 2025): quantized reasoners may generate more tokens (Lian et al., 2026), reason past correct intermediate answers (Lotfi et al., 2026), or change failure modes without a large accuracy drop (Oladri et al., 2026). Post-quantization adaptation then recovers accuracy at a fixed budget, by training adapters on a frozen quantized model (Dettmers et al., 2023), by jointly accounting for adapters and quantization (Xu et al., 2023; Jeon et al., 2025; Bondarenko et al., 2024), or by step-level SFT and QAT (Li et al., 2025; Lian et al., 2026). All of this work takes the decode budget as given.

Test-time scaling instead controls reasoning compute through repeated sampling or explicit token budgets. Self-consistency aggregates multiple reasoning traces (Wang et al., 2023), later work studies the non-monotonic relationship between chain-of-thought length and accuracy (Wu et al., 2025), and budget-forcing methods such as s1 make the thinking duration an explicit serving choice (Muennighoff et al., 2025). Model-dependent hardness is used the same way to curate SFT data (Lu et al., 2026) or to allocate reinforcement-learning rollouts (Zeng et al., 2025). Both lines select on an unquantized checkpoint, and to our knowledge no prior work tests whether the accuracy-maximizing budget or the hardness scores survive the precision change.

The location-scale map we fit in §5 is the two-parameter logistic form of item response theory (Birnbaum, 1968; Lord, 1980), chosen because it is the simplest global map under which labels could be corrected instead of remeasured. NLP work uses item response models to characterize benchmark items (Lalor et al., 2019; Vania et al., 2021). We instead hold the items fixed and fit one $(a, b)$ pair per intervention.

## 3 Method

We treat hardness as a property of a prompt–configuration pair. Let $c = (q, T, \pi)$ denote a configuration consisting of model precision $q$, decode budget $T$, and sampling protocol $\pi$. We use the

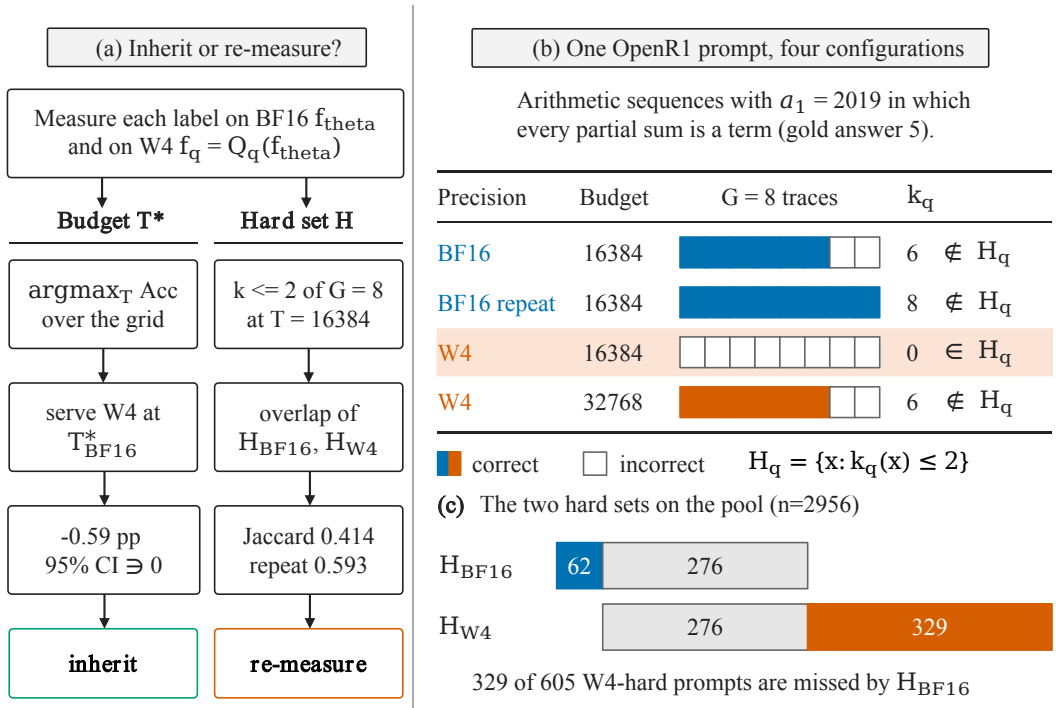



Figure 1: The audit (§3). (a) Each label is measured on BF16 and on W4 and the two are compared. Here $\mathcal{T}$=$\{8192, 16384, 32768\}$, the budget cost is paired over six GPQA seeds, and 0.593 is the BF16 repeat Jaccard. (b) One OpenR1 prompt: cells are the $G$=8 traces, filled when correct. The shaded row is the only hard-set member. The BF16 repeat shows sampling noise (6 vs. 8). The prompt is W4-hard at $T$=16384 but not at 32768. (c) The same comparison over the whole pool, aligned on the grey overlap.

conceptual decomposition

$$d(x; c) = d_{\text{intrinsic}}(x) + d_{\text{interaction}}(x, c). \tag{1}$$

The audit asks whether the interaction term is large enough to change two decisions: the accuracy-maximizing budget, and which prompts count as hard. Each is measured on BF16 and on the served model and the two are compared (Figure 1). The two tracks use separate data and produce separate outputs.

Let $f_\theta$ denote the BF16 source model. For $q \in \{\text{W4}, \text{INT3}\}$, the served model is $f_q = Q_q(f_\theta)$, where $Q_q$ is a frozen post-training quantization map (GPTQ unless stated, RTN in Appendix A.13). We write $f_{\text{BF16}} = f_\theta$ and take $Q_{\text{BF16}}$ to be the identity map. Let $\mathcal{T} = \{8192, 16384, 32768\}$, called the short, mid and long budgets (8k/16k/32k in figures), and $T_{\text{long}} = \max \mathcal{T}$. The candidate optimum is

$$T^*(q; \mathcal{D}) = \arg\max_{T \in \mathcal{T}} \text{Acc}_{\text{full}}(f_q, T; \mathcal{D}). \tag{2}$$

Each challenge has its own control. Budget selection from a precision-by-budget table in §3.1 addresses C1. After hard sets and the training schedule are defined in §3.2, the repeat reference of §3.3 addresses C2. The location-scale map for C3 is defined in §5.1. The workflow is sequential. The model is quantized once, the budget table and the hardness scores are measured on the same frozen model, and each output is compared with its BF16 counterpart and with a repeat. Algorithm 1 (Appendix A.1) states the procedure line by line, and caveats are in Appendix A.5.

### 3.1 Budget selection from a $q \times T$ table

Each prompt is decoded under every $(q, T)$ pair with the same budget prompt and a hard generation limit, recording correctness $c_{q,T}(x)$ and truncation $\tau_{q,T}(x)$. Each cell reports all-item accuracy

$\mathrm{Acc}_{\mathrm{full}}(q,T) = \frac{1}{|\mathcal{D}|}\sum_x c_{q,T}(x)$, with truncated traces counted as wrong, accuracy on finished traces $\mathrm{Acc}_{\mathrm{fin}}$, and the truncation rate $\mathrm{Trunc}$. We select $T^*$ on $\mathrm{Acc}_{\mathrm{full}}$ and name it only when long-budget truncation is below $5\%$, since a longer cap otherwise raises accuracy merely by letting more traces finish. Cells above the gate are reported as unnamed. On larger models whose long generations still truncate we close the thinking segment at $90\%$ of $T$ (measurement only). The budget cost of inheritance is W4 accuracy at the BF16-named $T^*$ minus W4 at its own $T^*$.

### 3.2 Hard sets and the training schedule

Hardness is scored at the same mid budget for every precision, so that precision is not confounded with budget. Budget dependence is measured separately by rescoring at the long budget. For each prompt $x$ we draw $G{=}8$ traces and define

$$k_q(x) = \sum_{g=1}^{G} c_q^{(g)}(x) \in \{0,\dots,G\}. \tag{3}$$

The hard set is $\mathcal{H}_q = \{x : k_q(x) \le 2\}$. We use $k_q \le 2$ because the stricter $k_q = 0$ subset is too small to support a training arm (Appendix A.10).

Let $\mathcal{C}_{\mathrm{tea}}$ contain prompts for which the BF16 teacher produces a correct chain of thought of at least $512$ tokens. The distillation-relevant hard set is

$$\mathcal{H}_q^{\mathrm{dist}} = \mathcal{H}_q \cap \mathcal{C}_{\mathrm{tea}}. \tag{4}$$

Prompts without a usable teacher trace cannot enter supervised fine-tuning. Since $\mathcal{H}_q^{\mathrm{dist}}$ does not record the scoring budget, we write $F_{q,T}$ for it when scored at budget $T$, abbreviated $F_{\mathrm{BF16}}$ and $F_{\mathrm{W4}}$ at the shared mid cap and $F_{\mathrm{mid}}$ and $F_{\mathrm{long}}$ when the precision is W4.

Hard-set oversampling (Fo) builds a training multiset $\mathcal{S}_q$ by sampling from $\mathcal{H}_q^{\mathrm{dist}}$ with replacement until it holds $N = |\mathcal{C}_{\mathrm{tea}}| = 2956$ examples, the number of teacher-correct traces in our OpenR1-Math-220k sample. We train LoRA-SFT on $\mathcal{S}_q$, merge the adapter and reapply GPTQ, and compare against random controls matched in unique prompts, example count and repetition rule. No evaluation item enters $\mathcal{S}_q$.

### 3.3 The reference for movement

Sampling alone changes hard-set membership, so every cross-configuration overlap is read against an independent repeat that holds $(q,T,\pi)$ fixed and changes only the random realization. The protocol $\pi$ uses temperature $1.0$, $\mathrm{top_p}{=}0.95$ and the budget prompt. Appendix A.11 gives the computational overhead.

## 4 Budget Choices Are Stable, Hardness Labels Are Not

We apply the two recalibration checks of §3 to the same BF16-to-W4 change. The budget argmax survives it (§4.1); the hardness label does not, and the SFT gain it appears to buy does not survive matched random controls (§4.2).

**Setup.** The source model is BF16 Qwen3.5-4B, compared at three precisions: **(1)** BF16, **(2)** GPTQ-W4 as the serving configuration, and **(3)** INT3 as a collapse probe (Appendix A.17). Hard sets are scored on $2956$ teacher-correct OpenR1-Math-220k prompts, which also form the SFT pool, and hardness-based oversampling (Fo) is compared with random controls matched in unique prompts and repetition (Ro and Random-355, Appendix A.15). GPQA-Diamond ($n{=}198$) is the primary benchmark and MATH-500 the secondary one. MMLU-Pro and code replicate the shape analysis, and AIME-2024 and GSM8K are smaller checks (Appendix A.14). Other models are budget-only checks; RTN-W4 (group size $128$) adds a six-seed budget sweep and one hardness repeat pair (Appendix A.13).

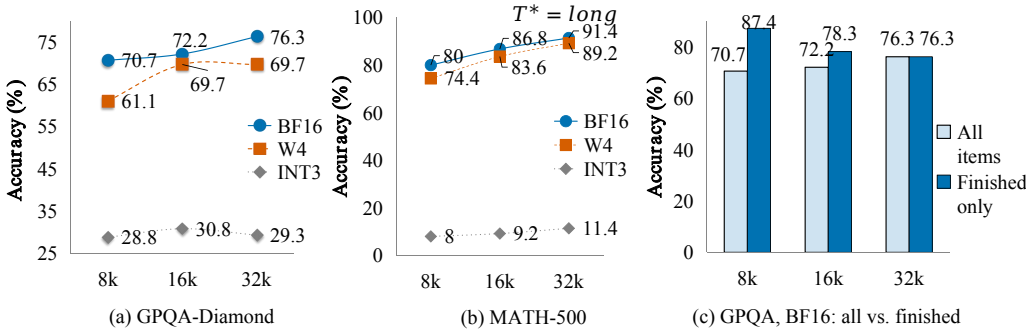



Figure 2: The decode budget transfers under quantization (Qwen3.5-4B, §4.1). (a,b) $\mathrm{Acc}_{\mathrm{full}}$ at 8k/16k/32k for BF16 (solid), W4 (dashed) and INT3 (dotted). (a) GPQA, seed 42, paired decode: BF16 peaks at long and W4 ties mid/long at 69.7. INT3 is flat and fails the truncation gate. (b) MATH: both usable precisions peak at long. (c) BF16 GPQA on all items (light) and on non-truncated traces (dark). Truncation is 39.9%, 9.1% and 0% at 8k, 16k and 32k (Table 6), so non-truncated accuracy falls with the cap (87.4→76.3) and would name the shortest cap. MATH and INT3 come from the unpaired sweep.

Table 1: Qwen3.5-4B/GPQA Acc–$T$ under six decode seeds with the decode stream paired across budgets ($n$=198). Counts are correct items. Mismatch is W4 accuracy at the BF16-named $T^*$ minus W4 at its own $T^*$. s/m/l: 8192/16384/32768.

| Seed | BF16 s/m/l | BF16 $T^*$ | W4 s/m/l | W4 $T^*$ / mismatch |
|---|---|---|---|---|
| 42 | 140/143/151 | long | 121/138/138 | mid / 0 pp |
| 43 | 132/149/156 | long | 125/134/138 | long / 0 pp |
| 44 | 138/147/154 | long | 111/148/141 | mid / −3.54 pp |
| 45 | 130/151/153 | long | 117/136/147 | long / 0 pp |
| 46 | 129/144/153 | long | 121/137/142 | long / 0 pp |
| 47 | 132/144/149 | long | 120/130/140 | long / 0 pp |

### 4.1 Budget selection on Qwen3.5-4B/GPQA

The budget argmax is largely preserved. With the decode random-number stream paired across budgets, so that a prompt meets the same sampling noise at every cap, BF16 selects long in all six seeds and W4 in four of six (Table 1). Serving W4 at the BF16-named cap costs −0.59 pp on average, with a paired item-bootstrap 95% interval of [−1.94, +0.76] pp (cells in Appendix A.3). Because each W4 optimum is chosen on the same seed it is scored on, this in-sample cost is conservative: picking W4's cap on one seed and scoring it on the other five, the BF16 cap is on average 0.89 pp *better* than W4's own choice. The gain comes from the two seeds where W4 names mid (42 and 44), whose choice loses 2.32 and 3.03 pp on the held-out seeds. In the other four the two caps coincide.

BF16's preference is stable, with a mid-minus-long margin of −2 to −9 items in every seed. W4 is noisier (−7 to +11 items long minus mid), and the only nonzero mismatch, −3.54 pp in seed 44, falls on a seed where W4's own argmax is unstable rather than on a wrong BF16 cap. Without paired streams an earlier sweep reported −2.4 ± 2.3 pp, which does not replicate (Appendix A.2), and selecting on non-truncated accuracy would name the shortest cap (Figure 2(c)).

The other Qwen results agree. MATH peaks at long in both precisions (Figure 2(b)), Qwen3.5-9B selects long at both precisions under force-close (138/157/160 vs. 120/136/150), and Qwen3.5-2B truncates at 7.1%/18.7% at long so $T^*$ stays undefined. The 4B and 9B results are robust to the gate between 5% and 7% (Appendix A.12) and to the budget prompt (Table 14). Phi-4-mini is the only model whose BF16 curve does not peak at long, but its curves are flat (pooled BF16 40.1/39.1/37.7%) and the gate names a BF16 cap in only one of three paired seeds, where W4 differs by −1.52 pp (3 items). It therefore gives no clean test of an interior optimum (Table 15). RTN-W4 truncates 17–22% of long traces, so no RTN budget is named (Appendix A.13).

Table 2: Hard-set overlap on the teacher-correct OpenR1 pool ($|\mathcal{C}_{\text{tea}}|$=2956). The same-precision repeats provide references for sampling variability. The last row is driven by truncation (§4.2).

| Comparison | $\lvert F_a\rvert$ | $\lvert F_b\rvert$ | Shared | Jaccard |
|---|---|---|---|---|
| BF16-mid vs. BF16-mid repeat | 338 | 344 | 254 | 0.593 |
| W4-mid vs. W4-mid repeat | 605 | 604 | 477 | 0.652 |
| BF16-mid vs. W4-mid | 338 | 605 | 276 | 0.414 |
| W4-mid vs. W4-long | 605 | 334 | 272 | 0.408 |

Table 3: GPQA correct counts at the long evaluation budget ($n = 198$). Random-355 matches Long-hard in unique-prompt count, training-example count, and repetition rule.

| Training data | Unique | Seed 42 | Seed 43 | Seed 44 | Mean |
|---|---|---|---|---|---|
| Full pool | 2956 | 131 | 136 | 136 | 134.3 |
| Mid-hard | 630 | 142 | 140 | 134 | 138.7 |
| Long-hard | 355 | 142 | 142 | 139 | 141.0 |
| Random-355 | 355 | 143 | 140 | 144 | 142.3 |
| $\Delta$ Long-hard − Full pool | — | +11 | +6 | +3 | +6.7 |
| $\Delta$ Long-hard − Random-355 | — | −1 | +2 | −5 | −1.3 |

### 4.2 Hard sets move across precision but not across budget

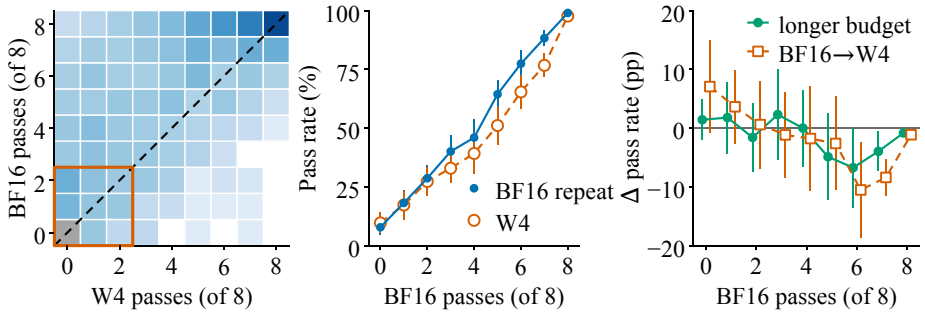
Hard-set membership moves across precision by more than sampling noise. On the teacher-correct OpenR1 pool ($|\mathcal{C}_{\text{tea}}| = 2956$), the BF16 and GPTQ-W4 mid-budget hard sets contain 338 and 605 prompts and share 276, a Jaccard overlap of $0.414$ with a paired item-bootstrap 95% interval of $[0.377, 0.451]$. An independent BF16 repeat yields a 344-prompt set overlapping the original at $0.593$ $[0.547, 0.640]$, and a W4 repeat yields 604 prompts overlapping at $0.652$ $[0.617, 0.686]$, so the cross-precision overlap falls below its own references by $18.0$ pp $[13.1, 22.9]$ and $23.8$ pp $[19.4, 28.1]$ (Table 2).

The gap is not an artefact of set size, accuracy drop or threshold. Comparing the 338 lowest-$k$ prompts of each arm gives $0.383$ against $0.589$ for the BF16 repeat. Simulating W4 as a pure logit shift of the BF16 counts gives $0.508$ $[0.485, 0.530]$, still well above $0.414$ (§5.4). For $k_{\max} = 0, \ldots, 4$ the overlaps range from $0.213$ to $0.557$, and none makes the sets interchangeable (Appendix A.10).

Across budget, by contrast, the movement is truncation. Rescoring the pool with W4 at the long budget gives an overlap of $0.408$ $[0.371, 0.445]$ with the mid set, but $68\%$ of the W4 mid hard set has at least one trace cut off at the mid budget, against $8\%$ at long. On the 1746 prompts untruncated in every arm, the W4 mid–long overlap is $0.639$, level with the repeats there (0.631–0.636), while BF16–W4 stays at $0.537$. The precision gap also survives where truncation is nearly absent: at the long budget only $0.4\%$ of the BF16 and $8\%$ of the W4 hard set is cut off ($44\%$ and $68\%$ at mid), yet the two share $0.482$ ($0.472$ size-matched), against $0.612$ for a BF16 long-budget repeat.

The cross-precision movement is not specific to GPTQ. An RTN-W4 hard set on the same pool (1258 prompts, mid budget) overlaps the BF16 set at $0.216$, against $0.774$ for an independent RTN repeat and $0.364$ $[0.351, 0.378]$ for the fitted map of §5.4, which accounts for the larger set size. We do not use RTN to identify the mechanism: $92\%$ of its hard set has at least one trace truncated at the mid budget (GPTQ: $68\%$), so its movement mixes any per-prompt interaction with the truncation confound above (Appendix A.13).

Training on the moved set does not reliably pay off. Against its matched random controls (Ro at mid, Random-355 at long), Fo wins two of three GPQA seeds at mid ($143/141/141$ vs. $139/129/145$) and one of three at long ($-1/+2/-5$ items, against $+11/+6/+3$ over the full pool; Table 3). The gain over the full pool is therefore a gain from a smaller, repeated set, which a random set of the same size and repetition reproduces. MATH moves by $+0.4$, $-0.8$ and $0.0$ pp. All nine paired intervals cross zero (smallest McNemar $p$=0.088). The training sets were selected before a re-grading that removed grader false negatives, so these runs test the allocation rule (Appendix A.4). With three seeds, no gain is detected at this power.




(a) Pass counts, BF16 vs. W4 (b) Loss by difficulty (c) Budget vs. W4

Figure 3: Why the labels move (§5). (a) Joint BF16/W4 pass counts ($G$=8, mid budget, $n$=2956, log scale). The box $k_q \leq 2$ selects 338 and 605 hard prompts sharing 276 (Jaccard 0.414 vs. 0.593 for a BF16 repeat). (b) Mean pass rate by BF16 pass count for a BF16 repeat (solid) and W4 (dashed), $n$=1746: the curves coincide for $k \leq 2$ and W4 loses 12–13 points at $k$=5–7 on average. This global trend is real but, as §5.4 shows, too small to explain panel (a). (c) Loss relative to a same-config repeat for two operations of equal effect size ($b \approx -0.27$), BF16 mid→long (solid) and BF16→W4 at long (dashed). Both are largest at $k$=5–7 and their scales differ by $-0.002$ $[-0.07, 0.07]$. Bars are 95% CIs. Group sizes and paired values are in Table 8.

## 5 What Moves the Labels: Per-Prompt Interaction, Not a Global Map

If quantization acted on hardness through one global map, a BF16 label could be transformed into a W4 label. We fit the simplest such map and show by forward simulation that it under-predicts the observed movement.

### 5.1 What a pure location shift would imply

We work on the latent hardness scale. For a prompt $x$ scored with $G$=8 traces as in §3, write $\hat{p}(x)$ for the shrunk pass rate under a beta-binomial prior fitted to the reference arm, and $z(x) = \operatorname{logit} \hat{p}(x)$ for its position on the latent scale (higher is easier). An intervention maps the reference arm to a target arm, and we summarize that map by two parameters,

$$\Pr[\text{correct} \mid x] \;=\; \sigma(a\, z(x) + b)\,, \tag{5}$$

where $\sigma$ is the logistic function, $b$ is a location shift, and $a$ is a scale. This is the two-parameter logistic model of item response theory (Birnbaum, 1968) with the reference arm supplying the latent scale in place of an estimated ability, so $a$ is a discrimination parameter that describes the intervention rather than the serving configuration.

Any map of this form preserves the expected hardness ordering for $a > 0$, so it cannot by itself reorder prompts. What it changes is how many prompts cross a fixed threshold ($b$) and how far apart they sit relative to the binomial noise of $G$ traces ($a$). If equation 5 with fitted $(a, b)$ reproduced the observed Jaccard, a BF16 label could be corrected by the map. If not, the residual is a prompt-specific term, the $d_{\text{interaction}}$ of equation 1. Since $a$ is estimated from pass counts, not set overlaps, it does not depend on the hard-set threshold; it is identified only when accuracy moves.

### 5.2 Compression across four operations

On the teacher-correct OpenR1 pool we compare two budget increases and two quantization steps on the $n = 1746$ prompts untruncated in every arm. Each contrast carries a calibration arm, an independent repeat of the reference where $a$=1 by construction, and the primary test is the paired difference $a_{\text{tgt}} - a_{\text{cal}}$, which cancels any bias the estimator shares between arms.

Table 4: Compression across operations on the OpenR1 teacher-correct pool, restricted to the $n = 1746$ prompts where no arm is truncated (Appendix A.6: the reference-side filter gives stronger compression, e.g. $0.791$ for the long quantization step). $a_{\mathrm{cal}}$: same estimator on an independent repeat of the reference. $Z$: $a_{\mathrm{tgt}}$ against a pure-shift null simulated at the row's own $b$. The first row does not move accuracy, so its $a_{\mathrm{tgt}} \approx 1$ is an identity.

| Operation | Kind | ΔAcc | $b$ | $a_{\mathrm{tgt}}$ | $a_{\mathrm{cal}}$ | $Z$ |
|---|---|---|---|---|---|---|
| W4 mid → W4 long | budget | $+0.0004$ | $+0.006$ | $0.994$ | $1.020$ | $+0.06$ |
| BF16 mid → BF16 long | budget | $-0.0119$ | $-0.257$ | $0.862$ | $0.978$ | $-4.75$ |
| BF16 long → W4 long | quant | $-0.0153$ | $-0.284$ | $0.864$ | $1.059$ | $-4.57$ |
| BF16 mid → W4 mid | quant | $-0.0276$ | $-0.571$ | $0.845$ | $0.978$ | $-5.32$ |

Table 4 reports the four contrasts. The three operations that move accuracy all compress the scale, with $a_{\mathrm{tgt}}$ between $0.845$ and $0.864$ against a null centered near $0.99$, and all three reject the pure-shift hypothesis at $Z \leq -4.57$. At $a \approx 0.86$ a prompt one latent unit above the mean loses roughly $0.14$ more logit than a prompt at the mean, so the loss concentrates on the easier side (Figure 3(b)). The row with $a_{\mathrm{cal}} = 1.059$ still rejects because the test is paired. The first row ($b = +0.006$) is uninformative rather than benign, and the BF16 budget increase compresses as much as quantization. Compression is not a math-grader artefact either. The same quantization contrast on MMLU-Pro (letter match, $n$=1924) and code (sandboxed execution, $n$=351) gives $a_{\mathrm{tgt}} = 0.815$ and $0.701$, with paired intervals excluding zero, and the verdicts are unchanged under probit and cloglog links (Appendix A.9).

### 5.3 Two controls on the shape estimate

If a jointly fitted $\hat{a}$ fell mechanically with $|b|$, every rejection above would merely restate the accuracy drop. We therefore simulate target arms from equation 5 with $a$=1 exactly, using the real reference arm, its $n$ and $G$, and sweep $b$ from 0 to $-1.2$, which covers every fitted shift in the paper. $\hat{a}$ drifts only from $0.992$ to $0.971$ ($0.021$), an order of magnitude below the observed $0.845$ to $0.864$. The same holds at the real effect size: a synthetic target built from BF16 by a pure logit shift that matches W4's accuracy ($0.815$ vs. $0.814$) returns $\hat{a} = 0.981$, against $0.681$ on the real W4 arm (math, mid budget, reference-side filter; Appendix A.9). Compression is therefore a property of the interventions and does not follow from their size.

Shape does not, however, identify *which* intervention was applied. At matched effect size, the BF16 budget increase and the long-budget quantization step differ in $a$ by $-0.002$ (paired $95\%$ interval $[-0.072, +0.067]$), so compression cannot be extrapolated from the kind of deployment change.

### 5.4 The fitted map does not reproduce the moved labels

Compression alone does not show that it is what moves the labels. We therefore run the map forward. Holding the reference pass counts fixed, we draw each prompt's latent from its beta-binomial posterior, simulate $G$=8 target traces under a given map, and recompute the $k \leq 2$ Jaccard (2000 draws, 300 per grid point for $s^*$). Three maps are compared: an independent repeat ($a$=1, $b$=0), a pure shift ($a$=1, $b$ fitted), and the fitted $(\hat{a}, \hat{b})$. To size the residual, we add per-prompt noise $\epsilon(x) \sim \mathcal{N}(0, s^2)$ to $\hat{a}z(x) + \hat{b}$ and report the $s^*$ at which the simulated Jaccard matches the observed one.

The simulator is calibrated, predicting $0.579$ and $0.671$ for the BF16 and W4 repeats against observed $0.593$ and $0.652$. Across precision, however, the fitted map predicts $0.523$ $[0.495, 0.551]$ against an observed $0.414$, no closer than the pure-shift $0.508$ (Table 5). Compression therefore explains little of the movement. Matching it requires a per-prompt interaction of $s^* \approx 1.0$–$1.2$ logit at the mid budget, which measures the scale of $d_{\mathrm{interaction}}$ in equation 1, against at most $0.65$ for any BF16 or W4 repeat, and this survives restriction to untruncated prompts, where the fitted map predicts $0.624$ $[0.567, 0.677]$ against an observed $0.537$ ($s^* = 1.03$). A stricter subset that also requires RTN to finish is inconclusive, because the simulator is miscalibrated there (Appendix A.13). At the long budget the excess is smaller ($0.75$). Across budget the pattern is reversed: on the full

Table 5: Forward simulation of the $k \leq 2$ hard-set Jaccard. Sim columns hold the reference counts fixed and draw the target from the stated map. $s^*$: s.d. (logit) of per-prompt interaction needed to match the observed Jaccard. "–" means the fitted map already predicts at or below the observation. BF16 and W4 repeats give $s^* \leq 0.4$ on the full pool and $\leq 0.65$ on the untruncated subset.

| Pool | Pair | Obs. | Pure shift | Fitted $(\hat{a}, \hat{b})$ | $s^*$ |
|---|---|---|---|---|---|
| Full (2956) | BF16 vs. W4, mid | 0.414 | 0.508 | 0.523 | 1.22 |
| | BF16 vs. W4, long | 0.482 | 0.565 | 0.556 | 0.75 |
| | W4 mid vs. long | 0.408 | 0.519 | 0.429 | 1.47 |
| Untrunc. (1746) | BF16 vs. W4, mid | 0.537 | 0.634 | 0.624 | 1.03 |
| | BF16 vs. W4, long | 0.559 | 0.637 | 0.615 | 0.75 |
| | W4 mid vs. long | 0.639 | 0.670 | 0.669 | 0.52 |
| | BF16 mid vs. long | 0.616 | – | 0.625 | 0.29 |

pool the W4 budget pair needs $s^* = 1.47$, but on untruncated prompts it needs $0.52$, and the BF16 budget pair $0.29$, both within the repeat range. The budget movement of §4.2 is truncation. Under GPTQ, the precision movement is a prompt-specific interaction that no global $(a, b)$ can absorb, so a BF16 hard set cannot be corrected into a W4 one and has to be remeasured.

**Why one measurement transfers and the other does not.** The two findings are consistent with a single picture. The budget argmax compares accuracies averaged over the evaluation set. A per-prompt term centered across prompts moves such an average little, and it enters W4's cells at every cap, so it largely cancels when caps are compared and the budget ordering survives. Hard-set membership instead thresholds each prompt separately, so the same term of about one logit moves every prompt near the threshold. We would therefore expect aggregate decisions to transfer and per-prompt decisions to require remeasurement whenever $d_{\text{interaction}}$ is large relative to repeat noise; for example, a router that thresholds prompts by BF16 difficulty should fail to transfer. This is an interpretation and a testable prediction, not a tested claim.

## 6 Conclusion

We asked which measurements taken on a BF16 checkpoint a quantized reasoning model can inherit, and found that the answer differs between the two we studied. The decode budget transfers: reusing it on GPTQ-W4 costs $-0.59$ pp ($[-1.94, +0.76]$). Prompt hardness does not: the cross-precision Jaccard of $0.414$ sits below the repeat reference ($0.593$) and below what a fitted global correction predicts ($0.523$), and RTN-W4 shows the same mismatch. Quantization compresses the hardness scale, but under GPTQ the remaining movement is a prompt-specific interaction of about one logit. Apparent movement across budgets, by contrast, is truncation and vanishes on finished traces. In practice, inherit the budget, remeasure hardness on the deployed model rather than correcting a BF16 label, and validate any hardness-based allocation against a matched random control. The same audit, with its truncation gate, repeat reference and forward simulation, applies unchanged to other quantizers and models.

**Limitations.** The main results come from one model (Qwen3.5-4B) and one quantizer (GPTQ-W4); RTN-W4 truncates too often to test the budget or isolate the interaction. BF16 picks the top of the budget grid, so transfer of an interior optimum is untested. The budget result bounds the cost of inheritance (up to $1.94$ pp) without showing it is zero. Hardness uses one repeat per precision, SFT uses three seeds, and $s^*$ sizes the residual without explaining it.

## AI Use Statement

We used generative AI tools for language editing, LaTeX formatting assistance, and literature search. All AI-suggested references were verified against original sources, and only citations confirmed to exist in the official proceedings or on arXiv were included in the bibliography. The authors designed the protocol, ran all experiments, selected the reported results, formulated the scientific claims,

and verified every number against the run logs. No synthetic data, evaluation labels, proofs, or reported measurements were generated by these tools. All scientific claims, experimental design, and reported measurements are the authors' own work and responsibility.

## ETHICS STATEMENT

This work studies post-training quantization and test-time budgets of publicly released language models on public reasoning benchmarks and training pools (GPQA-Diamond, MATH-500, OpenR1-Math-220k, MMLU-Pro, and others listed in §4). It does not involve human subjects or new data collection from people. The interventions are an evaluation protocol and a data-allocation rule on an already-quantized checkpoint. We report both all-item accuracy and truncation so that accuracy is not inflated by dropping unfinished traces.

## REPRODUCIBILITY STATEMENT

The deployment recipe (Qwen3.5-4B, GPTQ W4, group size 128, MMLU-Redux boundary-coreset calibration), the budget set $\{8192, 16384, 32768\}$, the long-truncation gate for naming $T^*$, the force-close rule on 9B/Phi, and the hardness rule $G{=}8$, $k_q \leq 2$ are specified in §3 and the setup of §4. Decoding uses eight NVIDIA H100 GPUs with Hugging Face inference, `gptq_torch`, temperature 1.0, top_p=0.95, and per-item seeds. Training hyperparameters and allocation-arm definitions are in Appendices A.17 and A.15. Per-item prediction logs and the paired-resample statistics reported in Appendix A.8 are released as part of the anonymous supplementary material. The shape analysis of §5 runs on those same per-item pass counts and needs no additional decoding. The estimator, the truncation filters, the per-family nulls, the effect-size sweep, and the matched-effect bootstrap are included in the released scripts with fixed seeds. Mathematics answers are graded by symbolic comparison with `sympy`, while the replications on MMLU-Pro and code use letter matching and sandboxed execution and therefore do not share that grader.

## REFERENCES

Allan Birnbaum. Some latent trait models and their use in inferring an examinee's ability. In Frederic M. Lord and Melvin R. Novick (eds.), *Statistical Theories of Mental Test Scores*, pp. 397–479. Addison-Wesley, Reading, MA, 1968.

Yelysei Bondarenko, Riccardo Del Chiaro, and Markus Nagel. Low-rank quantization-aware training for LLMs. *arXiv preprint arXiv:2406.06385*, 2024.

Mengzhao Chen, Wenqi Shao, Peng Xu, Jiahao Wang, Peng Gao, Kaipeng Zhang, and Ping Luo. EfficientQAT: Efficient quantization-aware training for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10081–10100, Vienna, Austria, 2025. Association for Computational Linguistics.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. LLM.int8(): 8-bit matrix multiplication for transformers at scale. In *Advances in Neural Information Processing Systems*, 2022.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, 2023.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *International Conference on Learning Representations*, 2023.

Hyesung Jeon, Yulhwa Kim, and Jae-Joon Kim. L4Q: Parameter efficient quantization-aware fine-tuning on large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2002–2024, Vienna, Austria, 2025. Association for Computational Linguistics.

John P. Lalor, Hao Wu, and Hong Yu. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4249–4259, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1434.

Zhen Li, Yupeng Su, Songmiao Wang, Runming Yang, Congkai Xie, Aofan Liu, Ming Li, Jiannong Cao, Yuan Xie, Ngai Wong, and Hongxia Yang. Quantization meets reasoning: Exploring and mitigating degradation of low-bit LLMs in mathematical reasoning. *arXiv preprint arXiv:2505.11574*, 2025.

Xinyu Lian, Walid Krichene, Beichen Huang, Masahiro Tanaka, Olatunji Ruwase, Li Zhang, and Minjia Zhang. Quantization inflates reasoning: Token inflation as a hidden cost of low-bit reasoning models. *arXiv preprint arXiv:2606.25519*, 2026.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of Machine Learning and Systems*, 2024.

Ruikang Liu, Yuxuan Sun, Manyi Zhang, Haoli Bai, Xianzhi Yu, Tiezheng Yu, Chun Yuan, and Lu Hou. Quantization hurts reasoning? an empirical study on quantized reasoning models. In *Second Conference on Language Modeling*, 2025.

Frederic M. Lord. *Applications of Item Response Theory to Practical Testing Problems*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1980.

Sanae Lotfi, Polina Kirichenko, Steven Li, and Zechun Liu. Quantized reasoning models think they need to think longer, but they do not. *arXiv preprint arXiv:2606.00206*, 2026.

Aojun Lu, Tao Feng, Hangjie Yuan, Wei Li, and Yanan Sun. Why does RL generalize better than SFT? a data-centric perspective on VLM post-training. *arXiv preprint arXiv:2602.10815*, 2026.

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, Suzhou, China, 2025. Association for Computational Linguistics.

Renuka Oladri, Mohan Vamsi Varadaraju Priya, and Jerry Wu. Silent failures in quantized LLM reasoning: A taxonomy-based analysis of hollow convergence and failure mode shifts. *arXiv preprint arXiv:2607.09999*, 2026.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *International Conference on Learning Representations*, 2024.

Clara Vania, Phu Mon Htut, William Huang, Dhara Mungra, Richard Yuanzhe Pang, Jason Phang, Haokun Liu, Kyunghyun Cho, and Samuel R. Bowman. Comparing test sets with item response theory. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1141–1158. Association for Computational Linguistics, August 2021. doi: 10.18653/v1/2021.acl-long.92.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. *arXiv preprint arXiv:2502.07266*, 2025.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, 2023.

Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, Xiaopeng Zhang, and Qi Tian. QA-LoRA: Quantization-aware low-rank adaptation of large language models. *arXiv preprint arXiv:2309.14717*, 2023.

Yongcheng Zeng, Zexu Sun, Bokai Ji, Erxue Min, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Haifeng Zhang, Xu Chen, and Jun Wang. CurES: From gradient analysis to efficient curriculum learning for reasoning LLMs. *arXiv preprint arXiv:2510.01037*, 2025.

# A Appendix

## A.1 Protocol in algorithmic form

**Algorithm 1** Precision-conditioned budget selection and hardness-based allocation

**Require:** BF16 source model $f_{\mathrm{BF16}}$, target precision $q$, PTQ map $Q_q$, fixed sampling protocol $\pi$, evaluation set $\mathcal{D}$, teacher-correct pool $\mathcal{C}_{\mathrm{tea}}$, training-example budget $N$
**Ensure:** serving choice $T^*(q)$ (or undefined) and training multiset $\mathcal{S}_q$ (or undefined)
1: $f_q \leftarrow Q_q(f_{\mathrm{BF16}})$
2: **for** $T \in \{8192, 16384, 32768\}$ **do**
3: decode $\mathcal{D}$ under $(q, T)$
4: record $\mathrm{Acc}_{\mathrm{full}}$, $\mathrm{Acc}_{\mathrm{fin}}$, and $\mathrm{Trunc}$
5: **end for**
6: **if** $\mathrm{Trunc}(q, T_{\mathrm{long}}) < 5\%$ **then**
7: $T^*(q) \leftarrow \arg\max_T \mathrm{Acc}_{\mathrm{full}}(q, T)$
8: **else**
9: $T^*(q) \leftarrow$ undefined
10: **end if**
11: **for** $x \in \mathcal{C}_{\mathrm{tea}}$ **do**
12: draw $G$=8 mid-budget traces on $f_q$
13: $k_q(x) \leftarrow \sum_g c_q^{(g)}(x)$
14: **end for**
15: $\mathcal{H}_q^{\mathrm{dist}} \leftarrow \{x \in \mathcal{C}_{\mathrm{tea}} : k_q(x) \le 2\}$
16: **if** $\mathcal{H}_q^{\mathrm{dist}} = \varnothing$ **then**
17: **return** $(T^*(q), \text{undefined})$
18: **end if**
19: $\mathcal{S}_q \leftarrow$ sample with replacement from $\mathcal{H}_q^{\mathrm{dist}}$ until $|\mathcal{S}_q| = N$
20: **return** $(T^*(q), \mathcal{S}_q)$

▷ Long-hard Fo repeats hardness scoring at $T_{\mathrm{long}}$; see §A.4.

**Line-by-line description.** Line 1 applies the frozen GPTQ map once, so every later measurement uses the same served model. Lines 2–4 build one row of the precision-by-budget table of §3.1, decoding the evaluation set at each cap with the paired stream of Appendix A.2 and recording all-item accuracy, finished-trace accuracy and truncation. Lines 6–10 apply the truncation gate. A budget is named only when fewer than $5\%$ of long-budget traces are cut off, and it is then the argmax of all-item accuracy. Lines 11–13 score hardness at the shared mid budget by counting correct traces out of $G$=8 for every teacher-correct prompt. Line 15 keeps the prompts with at most two correct traces as the hard set of §3.2. The branch at Line 16 returns no training set if the hard set is empty. All observed hard sets are nonempty, so this branch only defines behavior outside the measured configurations. Line 19 oversamples the hard set with replacement to the fixed example budget $N$, and Line 20 returns the budget and the training multiset. Per-item seeds are recorded as identifiers of the random-number realization.

**Complexity.** With $C_{\mathrm{dec}}(T)$ the cost of one decode at cap $T$, Lines 2–4 cost $O(|\mathcal{T}|\,|\mathcal{D}|\,C_{\mathrm{dec}}(T_{\mathrm{long}}))$, Lines 11–13 cost $O(G\,|\mathcal{C}_{\mathrm{tea}}|\,C_{\mathrm{dec}}(\mathrm{mid}))$, and Lines 15–19 cost $O(|\mathcal{C}_{\mathrm{tea}}| + N)$. Decoding dominates, and the audit adds one repeat of each measurement to supply the reference of §3.3. Measured costs are in Appendix A.11.

### A.2 PAIRING THE DECODE STREAM ACROSS BUDGETS

Budget selection compares one prompt evaluated under three caps, so any difference between caps is a mixture of the cap's effect and the sampling noise of the three independent generations. Our primary protocol removes the second term by pairing the decode random-number stream across caps: for a fixed decode seed, the same prompt is served from the same stream at short, mid, and long, so a prompt that is answered correctly at mid and not at long differs because of the cap and not because it drew a different sample. All numbers in §4.1, Table 1, Table 6, and Figure 2(a) come from this protocol (`paired_budget_rng`=true).

An earlier three-seed sweep drew each cell from an independent stream. On that sweep BF16's argmax moved between mid and long across seeds and the mean cost of inheriting the BF16 cap was $-2.4 \pm 2.3$ pp, with a bootstrap interval of $[-5.05, +0.34]$ pp that already contained zero. Under pairing, BF16's mid-minus-long margin is negative in all six seeds and the cost falls to $-0.59$ pp $[-1.94, +0.76]$. We read the earlier sweep as unpaired sampling noise rather than a precision effect, and we report only the paired protocol. The comparison is one reason we treat a three-seed argmax as insufficient evidence for a budget claim at all: the argmax is a discrete function of counts that differ by a handful of items, and Table 1 shows W4's long-minus-mid margin ranging from $-7$ to $+11$ items across seeds that are otherwise identical.

### A.3 CELL-LEVEL BUDGET SWEEPS

Table 6 lists the six seed-42 cells behind Figure 2(a,c).

Table 6: Truncation-aware GPQA evaluation on Qwen3.5-4B, seed 42 under the paired-decode protocol. Accuracies are percentages; $n_{\text{fin}}$ is the number of items the non-truncated column is computed on. The non-truncated column peaks at the short cap in both precisions, on barely half the items, which is why $T^*$ is read off the all-item column.

| Precision | Budget | All-item | Non-trunc. | $n_{\text{fin}}$ | Truncation |
|---|---|---|---|---|---|
| BF16 | short | 70.71 | 87.39 | 119 | 39.90 |
| BF16 | mid | 72.22 | 78.33 | 180 | 9.09 |
| BF16 | long | 76.26 | 76.26 | 198 | 0.00 |
| GPTQ-W4 | short | 61.11 | 82.57 | 109 | 44.95 |
| GPTQ-W4 | mid | 69.70 | 74.71 | 174 | 12.12 |
| GPTQ-W4 | long | 69.70 | 70.05 | 197 | 0.51 |

### A.4 HARDNESS-AWARE SFT VS. MATCHED RANDOM CONTROLS

The control that matters randomly selects the same number of unique prompts and applies the same repetition rule, so the two arms differ only in the selection rule: Ro for the mid-budget Fo arm, and a Random-355 arm for Long-hard, which oversamples the 355 prompts hard under W4 at the long scoring budget. A full-pool arm over all 2956 teacher-correct prompts provides context (Table 3). These checkpoints predate the offline ANTLR rejudgment and retain their original 630- and 355-prompt selections, so they test the original allocation rule rather than retraining on the corrected sets. Mid-hard and Long-hard differ in both unique-prompt count and repetition factor, so their gap cannot be attributed specifically to hard-set size, repetition, or the scoring budget. Generation statistics do not rescue the result either: Long-hard outputs are generally shorter than Random-355 outputs and all long-budget cells truncate at most $0.51\%$, so the control's competitive accuracy is not a truncation advantage. Paired statistics are in Appendix A.8 and the W4-mined variant in Appendix A.18; three seeds limit what these tests can resolve, but their direction agrees with the claim that selection by hardness has no stable advantage.

### A.5 METHOD CAVEATS AND ADAPTER CONTROLS

We do not introduce a new quantizer or training algorithm. We read $k_q$ on the vanilla $Q_q(f_\theta)$ student, whereas the Fo model is trained in BF16, merged, and re-quantized. The weights that score hardness are therefore the deployment point before allocation, the only student available at selection

time, and are not bit-identical to the final served checkpoint. Two adapter variants were tried as controls and are not part of the protocol. Their GPQA long-minus-mid differences are $+2/+5/+6$ items (budget-aligned) and $-3/+10/+1$ (stopping-modified) across seeds, so neither consistently restores the BF16-preferred budget.

### A.6 Truncation conditioning in the shape estimates

Fitting equation 5 requires deciding which prompts to keep when some traces hit the budget cap. The two defensible choices disagree, and §5 uses both.

Conditioning on the reference arms only keeps the larger pool but leaves residual truncation in the target arm. Requiring every arm to be untruncated removes that residual and makes contrasts share one subset, at the cost of a smaller and easier pool. We avoid the third option, conditioning on the target arm alone, because truncation is a consequence of the intervention: selecting on it opens a collider path between the intervention and prompt hardness.

Table 7: The same three operations under reference-side conditioning and under the all-arms filter of Table 4. Point estimates of $a_{\mathrm{tgt}}$ are filter-dependent and must not be compared across columns, but every contrast that moves accuracy compresses under both filters.

| | Reference-side | | All arms untruncated | |
|---|---|---|---|---|
| Operation | $n$ | $a_{\mathrm{tgt}}$ | $n$ | $a_{\mathrm{tgt}}$ |
| W4 mid → W4 long | 1773 | 0.998 | 1746 | 0.994 |
| BF16 mid → BF16 long | 2214 | 0.865 | 1746 | 0.862 |
| BF16 long → W4 long | 2915 | 0.791 | 1746 | 0.864 |

Table 7 reports both. The W4 budget row fits $a_{\mathrm{tgt}} \approx 1$ under either filter, but it also fits $b \approx 0$ under either filter, so it is uninformative for the reason given in §5.2 rather than evidence of a pure shift. The two contrasts that move accuracy compress under both filters. The largest filter sensitivity is the quantization step at the long budget, where the all-arms subset is restricted to prompts that the mid-budget arms already complete and is correspondingly easier; the paired $T_2$ statistic excludes zero in both cases. The all-arms filter used in the main text is therefore the conservative one for this contrast ($0.864$ against $0.791$).

### A.7 Per-group values behind Figure 3

Table 8 lists every point of Figure 3(b,c). Prompts are grouped by the pass count $k$ of the reference arm, over the $n$=1746 prompts untruncated in all arms. Each $\Delta$ is the mean over the group of the per-prompt difference in pass rate between target and same-config repeat, with 95% CI $\bar{x} \pm 1.96\, s/\sqrt{n}$. The CI describes the group mean, not the spread of individual prompts.

### A.8 Paired SFT statistics

Table 9 reports per-seed, per-benchmark paired bootstrap 95% confidence intervals and exact McNemar $p$-values for the matched SFT comparisons. Bootstrap uses $100{,}000$ resamples on the $198$ GPQA items or $500$ MATH items within each seed. The three seeds are not pooled because they share item-level hardness structure. McNemar tests use discordant pairs only.

With Bonferroni correction over the nine comparisons (threshold $p < 0.0056$), no comparison is significant. The uncorrected tests also do not reject equality at the conventional $0.05$ level.

### A.9 Cross-family replication

Table 10 reports the shape estimate of §5 on three task families with unrelated scoring. The paired $T_2$ intervals are $[-0.354, -0.260]$ on math, $[-0.226, -0.128]$ on MMLU-Pro, and $[-0.327, -0.111]$ on code, so compression replicates in every family. The null s.d. is simulated separately per family because its width depends on $n$, and the paired $T_2$ statistic differences the two arms within a family so that any shared estimator bias cancels.

Table 8: Per-group values of Figure 3, in percent. Left: panel (b), BF16 repeat and W4 at the mid budget, grouped by BF16 mid passes, with the paired W4−repeat difference. Right: panel (c), the two matched operations, each grouped by its own reference arm, so group sizes differ. $\Delta$: mean $\pm$ 95% CI half-width.

| | (b) W4 at mid budget | | | | (c) BF16 mid → long | | (c) BF16 → W4, long | |
|---|---|---|---|---|---|---|---|---|
| $k$ | $n$ | Repeat | W4 | $\Delta$ | $n$ | $\Delta$ | $n$ | $\Delta$ |
| 0 | 53 | 7.8 | 9.7 | $+1.9 \pm 5.2$ | 53 | $+1.4 \pm 3.4$ | 46 | $+7.1 \pm 7.9$ |
| 1 | 42 | 18.2 | 17.3 | $-0.9 \pm 7.7$ | 42 | $+1.8 \pm 6.1$ | 69 | $+3.6 \pm 6.2$ |
| 2 | 55 | 28.6 | 27.3 | $-1.4 \pm 6.9$ | 55 | $-1.6 \pm 5.8$ | 42 | $+0.6 \pm 7.5$ |
| 3 | 49 | 40.1 | 32.9 | $-7.1 \pm 8.0$ | 49 | $+2.3 \pm 7.7$ | 54 | $-1.2 \pm 7.3$ |
| 4 | 37 | 45.9 | 39.2 | $-6.8 \pm 8.7$ | 37 | $+0.0 \pm 6.5$ | 49 | $-1.8 \pm 8.9$ |
| 5 | 46 | 64.4 | 51.1 | $-13.3 \pm 7.0$ | 46 | $-4.9 \pm 7.3$ | 43 | $-2.6 \pm 8.0$ |
| 6 | 52 | 77.4 | 65.4 | $-12.0 \pm 6.6$ | 52 | $-6.7 \pm 6.7$ | 57 | $-10.5 \pm 8.1$ |
| 7 | 82 | 88.3 | 76.7 | $-11.6 \pm 4.7$ | 82 | $-4.0 \pm 3.4$ | 119 | $-8.4 \pm 3.1$ |
| 8 | 1330 | 99.0 | 97.7 | $-1.3 \pm 0.4$ | 1330 | $-0.8 \pm 0.5$ | 1267 | $-1.1 \pm 0.3$ |

Table 9: Per-seed paired bootstrap and exact McNemar tests. Differences and confidence intervals are in percentage points. Fo–Ro is evaluated at the mid budget; Long-hard–Random-355 is evaluated at the long budget.

| Comparison | Benchmark | Seed | $\Delta$ (pp) | 95% CI (pp) | McNemar $p$ |
|---|---|---|---|---|---|
| Fo–Ro | GPQA-mid | 42 | $+2.02$ | $[-3.54, +7.58]$ | 0.597 |
| Fo–Ro | GPQA-mid | 43 | $+6.06$ | $[-0.51, +12.63]$ | 0.088 |
| Fo–Ro | GPQA-mid | 44 | $-2.02$ | $[-8.08, +4.04]$ | 0.636 |
| Fo–Ro | MATH-mid | 42 | $+0.40$ | $[-2.20, +3.00]$ | 0.883 |
| Fo–Ro | MATH-mid | 43 | $-0.80$ | $[-3.20, +1.60]$ | 0.618 |
| Fo–Ro | MATH-mid | 44 | 0.00 | $[-2.20, +2.20]$ | 1.000 |
| Long-hard–Random-355 | GPQA-long | 42 | $-0.51$ | $[-6.06, +5.05]$ | 1.000 |
| Long-hard–Random-355 | GPQA-long | 43 | $+1.01$ | $[-5.05, +7.07]$ | 0.875 |
| Long-hard–Random-355 | GPQA-long | 44 | $-2.53$ | $[-8.59, +3.54]$ | 0.500 |

Table 10: The quantization contrast BF16 mid → W4 mid across task families, each with reference-side conditioning and its own same-precision calibration arm. A calibration fit must be judged against its own row: $a_{\mathrm{cal}}$ lies within the null on all three families ($|z_{\mathrm{cal}}| \leq 1.0$). The math row uses the reference-side filter for comparability with the other families and so differs from Table 4, which restricts to prompts untruncated in every arm.

| Family | $n$ | $\Delta$Acc | $a_{\mathrm{tgt}}$ | $a_{\mathrm{cal}}$ | null s.d. | $z$ |
|---|---|---|---|---|---|---|
| Math (OpenR1) | 2214 | $-0.0595$ | 0.681 | 0.990 | 0.023 | $-12.89$ |
| MMLU-Pro | 1924 | $-0.0173$ | 0.815 | 0.989 | 0.026 | $-6.38$ |
| Code | 351 | $-0.0242$ | 0.701 | 0.910 | 0.056 | $-4.17$ |

**Link sensitivity.** A pure location shift is defined relative to a latent scale, so rejecting it is meaningful only for a named link. We therefore refit $a$ on the math contrasts under logit, probit and cloglog, each with its own simulated pure-shift null (300 simulations). cloglog is the informative case because it is asymmetric. Table 11 shows that the verdict does not change with the link: quantization rejects a pure shift under all three, with $T_2$ intervals excluding zero, and the W4 budget pair does not. The compression is weakest under cloglog ($z = -4.5$) but remains well outside the null. As a further check, a synthetic target built from the BF16 arm by a known pure logit shift of the observed size ($d = -1.08$, giving accuracy 0.815 against the real W4 0.814) returns $\hat{a} = 0.981$, against 0.681 on the real W4 arm, so the deficit is not a response to the accuracy drop itself. The $z$ values differ slightly from Table 10 because the null is re-simulated.

Table 11: Shape estimate under three links on math (Qwen3.5-4B, GPTQ-W4, mid budget), each with its own pure-shift null. Reference-side filter as in Table 10.

| Contrast | Link | $a_{\mathrm{tgt}}$ | $a_{\mathrm{cal}}$ | $z$ | $T_2$ | 95% CI of $T_2$ |
|---|---|---|---|---|---|---|
| BF16 $\to$ W4, mid ($n$=2214) | logit | 0.681 | 0.990 | $-12.7$ | $-0.309$ | $[-0.359, -0.266]$ |
| | probit | 0.761 | 0.991 | $-11.5$ | $-0.230$ | $[-0.274, -0.187]$ |
| | cloglog | 0.870 | 0.993 | $-4.5$ | $-0.122$ | $[-0.175, -0.068]$ |
| W4 mid $\to$ long ($n$=1773) | logit | 0.998 | 1.018 | $+0.5$ | $-0.020$ | $[-0.065, +0.027]$ |
| | probit | 0.998 | 1.014 | $+0.6$ | $-0.016$ | $[-0.053, +0.022]$ |
| | cloglog | 1.005 | 1.025 | $+1.0$ | $-0.020$ | $[-0.059, +0.023]$ |

### A.10 HARDNESS HISTOGRAMS AND $k$=0 CONTROLS

This appendix reports two readings that must not be conflated.

**Diagnostic reading (GPQA-Diamond, $n$=198, long cap).** With $G$=8, mean pass rates are $0.763/0.725/0.314$ for BF16/W4/INT3. Classic buckets ($k$=8, mixed, $k$=0) are $110/68/20$, $93/87/18$, and $2/174/22$. The INT3 histogram of $k$ is $22, 44, 44, 37, 26, 10, 6, 7, 2$ for $k = 0, \ldots, 8$. These are the numbers behind Table 12. W4 erodes the easy set ($110 \to 93$) and INT3 collapses almost entirely into the medium bucket (174 of 198). This population selects no training prompt. Using the training threshold $k \leq 2$ on this same GPQA diagnostic gives Jaccard $0.636$ for BF16–W4 and $0.208$ for BF16–INT3. These values belong to the GPQA evaluation population at the long cap. They describe how the hardness ordering of the evaluation set changes with $q$; they do not select any training prompt.

**Training reading (OpenR1 pool, mid cap).** This is the reading used by Fo in §3. Before the teacher-correct restriction, the BF16–W4 hard-set Jaccard is $0.83$ on OpenR1. After restricting selection to $\mathcal{C}_{\mathrm{tea}}$, the BF16–W4 Jaccard is $0.414$, with 329 W4-only items, 276 shared items, and 62 BF16-only items (Figure 3(a)). No OpenR1 INT3 pass table is used in this analysis. Within the corrected W4-hard set, $k = 0$ accounts for $238/605 \approx 39\%$ of the prompts. The already-completed Fo checkpoint used the pre-rejudgment 630-prompt training selection; the threshold sensitivity below describes the corrected labels and is not a claim that Fo was retrained. Threshold sensitivity on $\mathcal{C}_{\mathrm{tea}}$ ($|\cdot|$=2956), reported as $|F_{W4}|$ / $|F_{\mathrm{BF16}}|$ / shared: $k \leq 0$ gives Jaccard $0.213$ ($238/98/59$); $k \leq 1$ gives $0.301$ ($443/201/149$); $k \leq 2$ gives $0.414$ ($605/338/276$); $k \leq 3$ gives $0.491$ ($760/458/401$); $k \leq 4$ gives $0.557$ ($922/588/540$). No tested threshold makes the two hard sets interchangeable.

**Offline rejudgment protocol.** We reran the same deterministic correctness predicate ("extract_math_answer", "normalize_math", and "math_equal") over the saved predictions after installing `antlr4-python3-runtime==4.11.1`. This makes the LaTeX parser available for every arm and removes parser-induced false negatives; the rejudgment changed 162, 169, 184, 182, and 208 labels from incorrect to correct for BF16-main, BF16-repeat, W4-main, W4-repeat, and W4-long, respectively, with no correct-to-incorrect flips. All hardness counts, threshold sensitivity values, and overlap figures in the main text use these corrected labels. The already-completed Long-hard and Random-355 SFT checkpoints retain their original 355-prompt training selections; they are downstream controls and were not retrained after this offline rejudgment.

Table 12: BF16-to-W4 hardness transitions on the 198 GPQA-Diamond evaluation items, scored with $G$=8 at the long cap. Rows are BF16; columns are W4. This is the diagnostic reading, not the mid-budget OpenR1 pool that Fo actually mines.

| **BF16 to W4** | **Easy** | **Medium** | **Hard** | **Total** |
|---|---|---|---|---|
| **Easy** | 83 | 27 | 0 | 110 |
| **Medium** | 10 | 52 | 6 | 68 |
| **Hard** | 0 | 8 | 12 | 20 |
| **Total** | 93 | 87 | 18 | 198 |

### A.11 BUDGET OVERHEAD AND FREE-GENERATION LENGTH

The asymptotic cost of Algorithm 1 is given after the algorithm in Appendix A.1. Table 13 reports the measured accuracy and mean generation length under inherited and precision-specific budgets.

Table 13: Accuracy and generation length under inherited and precision-specific budgets.

| Benchmark | Precision | Budget | All-item | Mean tokens |
|---|---|---|---|---|
| GPQA | BF16 | mid | 76.26 | 8,722.66 |
| GPQA | GPTQ-W4 | mid | 68.18 | 8,713.17 |
| GPQA | GPTQ-W4 | long | 70.71 | 9,823.00 |
| MATH | BF16 | mid | 86.80 | 4,669.15 |
| MATH | GPTQ-W4 | mid | 83.60 | 4,959.40 |
| MATH | GPTQ-W4 | long | 89.20 | 6,052.84 |

Table 14 repeats the comparison without the budget prompt.

Table 14: Free-generation length and chain token inflation ratio (CTIR), the mean generated tokens at precision $q$ divided by the BF16 mean on the same benchmark. This protocol removes the budget prompt but still applies a 32k hard cap, and it does not eliminate truncation ($14.4\%$ BF16 / $12.2\%$ W4 on MATH), so it is a weaker-constraint comparison rather than an unconstrained one. W4 does not inflate; INT3 lengthens as accuracy collapses.

| Benchmark | Precision | All-item (%) | Mean tokens | CTIR |
|---|---|---|---|---|
| GPQA | BF16 | 78.79 | 11,490.80 | 1.000 |
| GPQA | GPTQ-W4 | 74.24 | 11,270.91 | 0.981 |
| GPQA | INT3 | 29.29 | 13,809.54 | 1.202 |
| MATH | BF16 | 81.20 | 12,037.65 | 1.000 |
| MATH | GPTQ-W4 | 81.00 | 11,524.94 | 0.957 |
| MATH | INT3 | 9.00 | 16,798.05 | 1.396 |

### A.12 EXTRA ACC–$T$ RUNS THAT WE DO NOT MIX

**9B without force-close.** Natural long truncation is $12.1\%$ (BF16) and $8.1\%$ (W4), with counts $129/156/157$ and $102/122/148$. Both precisions still select long, but the truncation gate prevents us from reporting $T^*$. Table 15 uses only the force-close run ($138/157/160$ and $120/136/150$). Force-close therefore decides only whether an argmax may be reported, not which budget is selected. W4's mid-to-long margin is larger under force-close (14 vs. 3 items), yet the argmax does not flip.

**Phi / 9B decoding.** Phi and 9B are not decoded like 4B: both use force-close, Phi's W4 long cells close $86$–$89\%$ of their traces that way, and its named cells still truncate at $16$–$26\%$ because the gate only inspects the long cap. Cross-model shape comparisons are therefore weaker than the within-model 4B comparison. Phi Acc is only $\sim$30–$44\%$, and its short/mid/long cells differ by a few items. Without the gate, Phi-4-mini seeds $42$ and $44$ would name short at both precisions, and seed $43$ would still disagree.

**Gate sensitivity.** Of the six Qwen long cells, four fall at $0.0$–$4.5\%$ and two at $7.1\%/18.7\%$ (2B), so any threshold in $[5\%, 7\%]$ partitions the Qwen evidence identically. Phi/BF16 seed $42$ ($5.1\%$) sits at the gate. Tightening to $3\%$ would additionally unname 9B/W4 ($4.5\%$) and Phi/BF16 seed $43$ ($4.0\%$). Loosening to $10\%$ would admit 2B/BF16 ($7.1\%$) while still excluding 2B/W4 ($18.7\%$), which would support only a one-sided comparison. We chose $5\%$ as the widest gate that keeps every admitted comparison two-sided.

### A.13 SECOND QUANTIZER: RTN-W4

RTN-W4 rounds Qwen3.5-4B weights to $4$ bits with group size $128$ and no calibration; weights are stored fake-quantized in BF16. It is much more degraded than GPTQ-W4 (pooled GPQA long $63.0\%$ vs. BF16 $77.1\%$) and truncates far more often.

Table 15: GPQA argmax on the shared 8k/16k/32k grid. $T^*$ is reported only if long truncation is $<5\%$. Counts are correct items out of $n=198$. Qwen3.5-9B names the same cap at both precisions and Qwen3.5-2B remains unnamed. The Qwen3.5-4B row is seed 42 of the paired-decode protocol, where W4 ties between mid and long (all six seeds are in Table 1). Phi-4-mini rows are paired-decode seeds; its BF16 cap is named only in seed 43, where the two precisions differ.

| Model | $q$ | short/mid/long | $T^*$ | long trunc |
|---|---|---|---|---|
| Qwen3.5-4B | BF16 | 140 / 143 / 151 | long | 0.0% |
| | W4 | 121 / 138 / 138 | mid[§] | 0.5% |
| Qwen3.5-9B[†] | BF16 | 138 / 157 / 160 | long | 3.5% |
| | W4 | 120 / 136 / 150 | long | 4.5% |
| Phi-4-mini[†] s42 | BF16 | 74 / 67 / 73 | unnamed | 5.1% |
| | W4 | 70 / 69 / 65 | short | 2.5% |
| Phi-4-mini[†] s43 | BF16 | 77 / 85 / 70 | mid | 4.0% |
| | W4 | 63 / 60 / 62 | short | 2.0% |
| Phi-4-mini[†] s44 | BF16 | 87 / 80 / 81 | unnamed | 7.6% |
| | W4 | 63 / 60 / 59 | short | 1.5% |
| Qwen3.5-2B[‡] | BF16 | 84 / 98 / 114 | unnamed | 7.1% |
| | W4 | 69 / 69 / 82 | unnamed | 18.7% |

[†]force_close_frac=0.9 so the long cap binds. [‡]Reproduction run. The grid was run in full, but both long cells exceed 5%, so $T^*$ is left undefined. [§]Exact tie at 138 items; the tie-break rule takes the first maximum in short/mid/long order. The gate tests long truncation only: Phi's named cells still truncate at 16.2% (BF16 mid, s43) and 22.7–26.3% (W4 short).

**Budget.** Table 16 uses the same six paired-decode seeds as Table 1. Every RTN long cell fails the 5% gate, so no RTN $T^*$ is named. Without the gate RTN peaks at long in five seeds and at mid in seed 47 (127 vs. 126), so serving RTN at the BF16 cap costs $-0.08$ pp on average. Because the ungated optimum sits at the top of the grid, this is not evidence that an interior optimum transfers.

Table 16: RTN-W4 GPQA counts on Qwen3.5-4B ($n=198$), paired-decode seeds. BF16 counts are in Table 1. The argmax is ungated; every long cell fails the 5% gate.

| Seed | RTN s/m/l | Long trunc. | Ungated argmax |
|---|---|---|---|
| 42 | 99/109/133 | 19.2% | long |
| 43 | 100/112/123 | 16.7% | long |
| 44 | 96/120/127 | 20.2% | long |
| 45 | 98/110/119 | 21.7% | long |
| 46 | 100/118/120 | 20.2% | long |
| 47 | 96/127/126 | 17.2% | mid |

**Hardness and forward simulation.** Table 17 repeats the simulation of §5.4 with RTN arms ($k \leq 2$, mid budget, $G=8$; RTN main and repeat use independent seeds). On the full pool, 92% of the RTN hard set is truncated, so the large $s^*$ for RTN pairs is dominated by truncation and we do not interpret it as an interaction scale. The 859-prompt subset is untruncated in all five arms. It is small (hard sets of 60–76 prompts), selects prompts that RTN finishes, and the simulator over-predicts the RTN repeat there (0.610 vs. 0.522), so it does not identify the interaction either. The primary untruncated GPTQ test is the 1746-prompt subset of Table 5.

### A.14 GSM8K IS SATURATED

We ran the same three-cap grid on GSM8K before selecting GPQA and MATH. Both precisions exceed 93% at the short cap and improve by less than one point across the grid (Table 18). We therefore do not interpret the discrete argmax as a meaningful value of $T^*$ on this saturated benchmark.

### A.15 DATA ALLOCATION ARMS

Table 19 defines every allocation arm used in §4 and in the exploratory comparisons of Appendix A.16. The arms differ only in which prompts receive the training-example budget. A is

Table 17: Hard-set Jaccard with RTN-W4 arms: observed, pure-shift and fitted-$(\hat{a}, \hat{b})$ simulations (95% interval), and $s^*$ as in Table 5. Trunc.: share of the second arm's hard set with at least one truncated trace.

| Pool | Pair | Obs. | Shift | Fitted | $s^*$ | Trunc. |
|---|---|---|---|---|---|---|
| Full (2956) | BF16 vs. BF16 rep. | 0.593 | 0.581 | 0.585 [0.543, 0.626] | – | 44% |
| | RTN vs. RTN rep. | 0.774 | 0.797 | 0.795 [0.775, 0.812] | 0.56 | 93% |
| | BF16 vs. RTN | 0.216 | 0.315 | 0.364 [0.351, 0.378] | 3.20 | 92% |
| | GPTQ vs. RTN | 0.398 | 0.511 | 0.539 [0.521, 0.556] | 2.05 | 92% |
| Untrunc. (859) | BF16 vs. BF16 rep. | 0.658 | 0.666 | 0.655 [0.560, 0.753] | – | 0% |
| | RTN vs. RTN rep. | 0.522 | 0.604 | 0.610 [0.516, 0.705] | 1.11 | 0% |
| | BF16 vs. RTN | 0.479 | 0.653 | 0.615 [0.519, 0.711] | 1.35 | 0% |
| | GPTQ vs. RTN | 0.485 | 0.698 | 0.654 [0.563, 0.747] | 1.50 | 0% |
| | BF16 vs. GPTQ | 0.609 | 0.665 | 0.660 [0.579, 0.746] | 0.71 | 0% |

Table 18: GSM8K all-item acc. (%). $\Delta_T \lesssim 1$pp, so this suite cannot identify $T^*$.

| $q$ | short | mid | long | $T^*$ |
|---|---|---|---|---|
| BF16 | 95.0 | 95.2 | 95.8 | unnamed |
| GPTQ-W4 | 93.4 | 93.7 | 94.3 | unnamed |

the untrained W4 model, B trains on the full teacher-correct pool, F trains on the W4-hard slice without oversampling, and Fo samples that slice with replacement until it contains $|\mathcal{C}_{\text{tea}}|$ training examples. Fo-long instead uses the long-hard slice obtained by rescoring the full teacher-correct pool at the long budget ($|F_{\text{long}}|$=355); Ro-355 is its size-matched random control. Ro and RoL are the size- and length-matched controls at the mid budget, and Dh is the DC-SFT sign flip that drops the W4-hard items instead of up-weighting them. Fo and Ro have the same training-example budget, unique-prompt count, and repetition factor, but their mean $k_{W4}$ values are 0.83 and 5.57. This comparison controls for size and repetition while changing the hardness of the selected prompts.

Table 19: Allocation arms. The training-example budget is $N = |\mathcal{C}_{\text{tea}}|$ for B, Fo, Ro, and RoL.

| Arm | Selection | $N$ | Unique | Mean $k_{W4}$ |
|---|---|---|---|---|
| A | none (vanilla W4) | – | – | – |
| B | all teacher-correct | 2956 | 2956 | 5.56 |
| F | $k_{W4} \leq 2$ at mid | 630 | 630 | 0.84 |
| Fo | $F$ oversampled to $N$ | 2956 | 630 | 0.83 |
| Fo-long | $k_{W4} \leq 2$ at long on full pool | 2956 | 355 | 1.16 |
| Ro | random $\|F\|$ subset, oversampled | 2956 | 630 | 5.57 |
| Ro-355 | random 355-prompt subset, oversampled | 2956 | 355 | 5.65 |
| RoL | length-matched $\mathcal{C}_{\text{tea}} \setminus F$ | 2956 | 630 | 5.94 |
| Dh | $k_{W4} > 2$ at mid | 2326 | 2326 | 6.84 |

In Table 3, "Full pool" is arm B, "Mid-hard" is Fo, "Long-hard" is Fo-long, and "Random-355" is Ro-355. Mean $k_{W4}$ for Fo-long and Ro-355 is measured at the mid budget so that it remains comparable with the other arms. Fo-long and Ro-355 are independently resampled for each SFT seed; values shown are seed 42 (Fo-long 1.16–1.18, Ro-355 5.53–5.82 across seeds 42–44).

### A.16 Small-pool and bucket-dropping negatives

**Small-pool replication on MATH-train.** An earlier replication mines a different and easier pool: MATH train with MATH-500 removed, sampled to 3000 prompts, which yields 2089 teacher-correct traces and only 154 W4-hard items. The two pools are disjoint by construction, since the OpenR1 sample used in §4 explicitly excludes every prompt from this earlier pool; the main result is therefore not a rescoring of the same questions. MATH train is easier for this student, with mean $k_{W4} \approx 7.07$ compared with 5.56 on OpenR1. Its hard rate is $154/2089 \approx 7\%$, compared with $630/2956 \approx 21\%$ on OpenR1. Matching the training-example budget requires repeating the same 154 prompts about $13.6\times$, instead of $4.7\times$ in the main experiment. On that pool Fo does not beat B on MATH non-

truncated accuracy (91.0 against 92.0) and is clearly worse on GPQA (66.7 against 73.2), although MATH all-item accuracy is higher (87.4 against 85.8). This exploratory result provides another case in which hard-prompt oversampling does not outperform the available baselines. Because pool composition, hard-set size, and repetition differ simultaneously, it does not identify which factor explains the outcome.

**Bucket-dropping controls.** Three arms keep or drop a single hardness bucket instead of reweighting by $k_{W4}$: De keeps the pool minus the easy items, M keeps only the medium items, and Dh keeps $k_{W4} > 2$ (Table 19). These arms run on the same 2089-trace MATH-train pool as the replication above, so their absolute numbers are not comparable to the main tables. Relative to B on that pool (MATH all-item / non-trunc. 86.4/90.5): De 83.8/88.8; M 86.0/90.5; Dh 85.6/90.9. Dropping easy or hard buckets does not beat full teacher-correct distillation.

### A.17 HYPERPARAMETERS

**Backbone.** Qwen3.5-4B main; 9B / 2B / Phi-4-mini only for the Acc–$T$ shape table.

**GPTQ.** Weight-only, symmetric, bits=4, group_size=128. Calibration is a 512-example MMLU-Redux-2.0 boundary coreset scored with Qwen3.5-4B: from a 5330-item pool we take the 256 items of smallest $|\text{margin}|$ and the 256 of smallest confusion gap, with no per-subject quota, which covers 54 of 57 subjects. Calibration text uses the chat template with an `ANSWER: <letter>` target and an empty thinking block, truncated at 2048 tokens; these are short multiple-choice items (median ≈87 words), not packed 2048-token sequences. INT3 is a collapse probe and uses the same coreset at bits=3. RTN-W4 uses bits=4, group_size=128, no calibration (Appendix A.13). Qwen3.5-9B and Phi-4-mini reuse the 4B-selected coreset with their own tokenizers; Qwen3.5-2B uses a coreset of the same size and construction mined on 2B. We separately ran a 512-segment WikiText-2 GPTQ ablation, but it does not enter any table in this paper.

**Budgets.** The candidate limits are 8192, 16384, and 32768 tokens, each used with the same budget prompt. The default cross-precision hardness probe uses the mid budget with $G$=8. The long-hard arm instead rescores the full teacher-correct pool ($|\mathcal{C}_{\text{tea}}|$=2956) at the long budget. We report $T^*$ only when long-budget truncation is below 5%. Qwen3.5-9B and Phi-4-mini use force_close_frac=0.9.

**Fo allocation.** $|\mathcal{C}_{\text{tea}}| = 2956$, $|F| = 630$, and hard is defined as $k_{\text{W4}} \leq 2$. The matched training-example budget is $N = |\mathcal{C}_{\text{tea}}|$, and SFT max_seq_len=8192. The multi-seed protocol is $\{42, 43, 44\}$; seed 45 is a GPQA-only confirmation run (Appendix A.18). Default allocation tables use the mid probe. Long-hard Fo uses $|F_{\text{long}}|$=355 oversampled to the same $N$; its size-matched random control Ro-355 draws 355 teacher-correct prompts and applies the identical repetition rule. GPQA long-budget evaluation of B, mid-hard Fo, long-hard Fo, and Ro-355 is in Table 3.

**Decode.** Eight NVIDIA H100 GPUs, Hugging Face; `GPTQMODEL_BACKEND=gptq_torch`; temperature 1.0; top_p=0.95; per-item seed.

### A.18 SEED-LEVEL FO RESULTS

Table 20 compares Fo mined at W4 and at BF16 on seed 42, where W4 mining leads by +4.0 pp on GPQA. Table 21 varies the SFT oversample seed while keeping the GPQA decode seed at 42. The sign is not stable, so we do not claim a mining-precision gain. Table 22 gives the per-seed Fo−B differences. Seed 45 is a GPQA-only confirmation (143 against 137 of 198). Its +3.03 coincides with seed 42 in item counts, although the two runs truncate differently (5.1%/6.6% against 2.5%/3.0%). The MATH sign is not consistent across seeds, so we do not claim a MATH improvement.

On AIME-2024 ($n$=30, mid, all-item / non-trunc. / Trunc), A is 40.0/100.0/63.3, B 40.0/92.3/56.7 and Fo 43.3/76.5/43.3. At this size the result is directional only.

Table 20: Matched seed-42 mid comparison (all-item / non-trunc. / Trunc, %). Non-truncated columns use different surviving sets.

| Hardness used for Fo | MATH | GPQA |
|---|---|---|
| $\text{Fo}_{W4}$ | 87.20 / 90.25 / 3.60 | 72.22 / 74.47 / 5.05 |
| $\text{Fo}_{BF16}$ | 84.80 / 86.71 / 2.20 | 68.18 / 69.23 / 1.52 |
| $\Delta$ all-item / $\Delta$ non-trunc. | $+2.4$ / $+3.5$ | $+4.0$ / $+5.2$ |

Table 21: GPQA mid correct counts ($n$=198). SFT oversample seed varies; decode seed is 42.

| SFT seed | $\text{Fo}_{W4}$ ($n$/trunc) | $\text{Fo}_{BF16}$ ($n$/trunc) | $\Delta n$ |
|---|---|---|---|
| 42 | 143 / 5.05% | 135 / 1.52% | $+8$ |
| 43 | 141 / 4.55% | 142 / 4.04% | $-1$ |
| 44 | 141 / 4.04% | 139 / 2.53% | $+2$ |

Table 22: All-item accuracy differences between Fo and B, in percentage points. Seed 45 is a GPQA-only confirmation run.

| Benchmark | 42 | 43 | 44 | 45 |
|---|---|---|---|---|
| MATH | $+2.00$ | $+0.20$ | $-0.80$ | — |
| GPQA | $+3.03$ | $+1.52$ | $+3.54$ | $+3.03$ |