SHORTER REASONING, EARLIER ANSWERS? AN EVALUATION OF REASONING INTERFACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models often reason at length before answering, increasing cost and latency. We evaluate two reasoning interfaces in different model families under a shared design. Qwen3 receives concision and budget instructions, and gpt-oss uses its trained reasoning effort settings. These interfaces can shorten traces, but completed responses alone do not establish whether an interface improves earlier answers or changes when the model finishes. We evaluate paired runs at shared reasoning allowances and distinguish the answers models generate from the answers we read off their unfinished reasoning. The main comparisons cover Qwen3-14B and gpt-oss-20b/-120b on 198 GPQA Diamond and 500 MMLU-Pro questions. Across seven shared allowances, low gpt-oss effort improves generated-answer accuracy over high effort by 4.5–18.9 percentage points at 512 tokens, while high effort is 4.4–8.8 points more accurate at 8,192 tokens. The early advantage is concentrated in pairs where lower effort has completed and high effort remains active. For Qwen3-14B on MMLU-Pro, a concise instruction improves generated-answer accuracy by 3.5 points at 512 tokens and 2.5 points at 2,048. On pairs that both policies leave unfinished at 512 tokens, it still improves generated answers by 4.1 points. A budget instruction that announces the allowance shortens reasoning but gives smaller and less consistent gains, and matching the announced allowance to the evaluation horizon adds little in the tested comparisons. Neither the effort advantage nor the concise gain reflects adaptation to a stated allowance, and neither establishes uniformly better unfinished reasoning.

1 INTRODUCTION

Large language models often solve hard questions by writing intermediate reasoning before giving a final answer, a behavior that began as chain-of-thought prompting and is now trained into models whose reasoning length or effort can be adjusted by the user (Wei et al., 2022; Kojima et al., 2022; Jaech et al., 2024; Guo et al., 2025). Accuracy on competition mathematics, graduate science, and code often grows as models spend more reasoning tokens during generation, sampling, or aggregation (Wang et al., 2022; Snell et al., 2024; Muennighoff et al., 2025; Aggarwal et al., 2026). This reasoning is costly and variable. A single answer can consume tens of thousands of tokens, and additional reasoning can yield little improvement in final accuracy (Chen et al., 2024; Ballon et al., 2025; Su et al., 2025; Sun et al., 2025a). Controlling that cost is therefore a practical constraint when models must answer within a limited inference budget (Sun et al., 2025b).

We use the term *reasoning control* for an intervention that can change how much intermediate reasoning a model produces before its final answer. Examples include training that favors shorter reasoning (Aggarwal & Welleck, 2025; Han et al., 2025; Arora & Zanette, 2025), settings such as gpt-oss reasoning effort (Agarwal et al., 2025), methods that steer generation (Li et al., 2025; Sun et al., 2025b; Wen et al., 2025), and prompts that request concise reasoning (Renze & Guven, 2024; Nayab et al., 2024; Xu et al., 2025). These controls reach the model through different *reasoning interfaces*: instructions written into the prompt or settings trained into the model. Final accuracy against realized token cost measures a useful efficiency outcome (Sui et al., 2025), but it does not establish what answer the model can provide after a fixed amount of reasoning, or whether a shorter trace reflects a change in the reasoning preceding the answer.

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

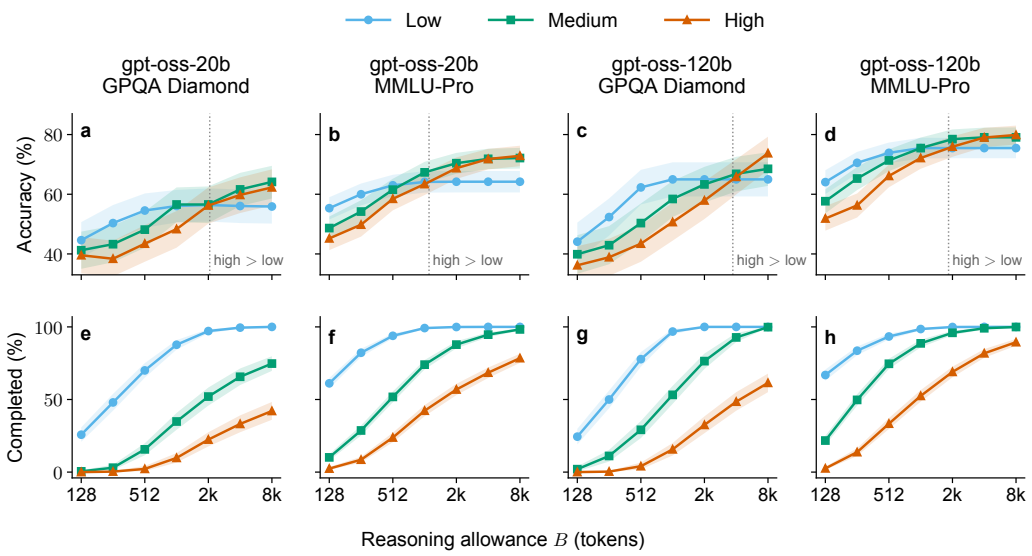


Figure 1: **The effort comparison changes with a shared reasoning allowance. a–d**, Source-policy cap-then-answer accuracy. **e–h**, Natural completion probability. Every curve contains all 198 GPQA Diamond or 500 MMLU-Pro items and three independent generations per effort. Already-completed runs retain their original answer. Active runs are interrupted and receive a greedy final answer with allowance $R = 16$. Parsing failures count as incorrect. Dotted guides in **a–d** mark the allowance at which high effort overtakes low effort, interpolated on the log axis between adjacent grid points. Shading gives pointwise 95% intervals from 10,000 paired item-bootstrap draws, averaging replicates within item and retaining each full horizon curve. Paired differences and simultaneous bands are in Table A4.

A model that stops early has a completed answer while another may still be working, which can be a useful improvement in answer delivery. The further scientific question is what the comparison establishes about the unfinished reasoning. A policy can change when it finishes, what it answers when interrupted, or both. Likewise, an instruction can shorten reasoning without helping the model adapt to an announced deadline. We therefore compare answers under the same imposed allowance, locate differences by completion state, and test whether matching an announced token count adds anything beyond asking for a different reasoning strategy.

We study Qwen3-14B prompting and gpt-oss-20b/-120b effort on GPQA Diamond (Rein et al., 2024) and a stratified 500-item MMLU-Pro sample (Wang et al., 2024), asking when each interface finishes with a usable answer and whether it adapts to an announced allowance. The paper makes three contributions. First, generated-answer curves across seven shared allowances reverse the ranking that completed responses suggest. High gpt-oss-120b effort reaches 79.3% on GPQA Diamond against 65.0% for low effort at natural completion, yet at a 512-token allowance low effort leads 62.3% to 43.4%, and high effort overtakes it only as the allowance grows (Figure 1). Second, a completion-stratum decomposition locates that early advantage in pairs where lower effort has completed while high effort remains active, without isolating a stopping-only mechanism. Third, an announced-budget matching test in Qwen3-14B separates the concise instruction, which improves early answers, from matching the stated number, which adds little, and both results hold on 500 further questions not used to choose the prompts. Both interfaces shift when the model finishes, while matching an announced number adds at most about a point to Qwen's accuracy and adding budget wording to gpt-oss effort gives no broad gain at 512 tokens. Each interface is studied in a different model family, so the two are compared as case studies rather than against each other.

2 Related Work

Strict output-budget evaluation and budget-dependent model or prompt choices are studied by Sun et al. (2025b), while Wu et al. (2025) examine interruption and difficulties during finalization. Datta

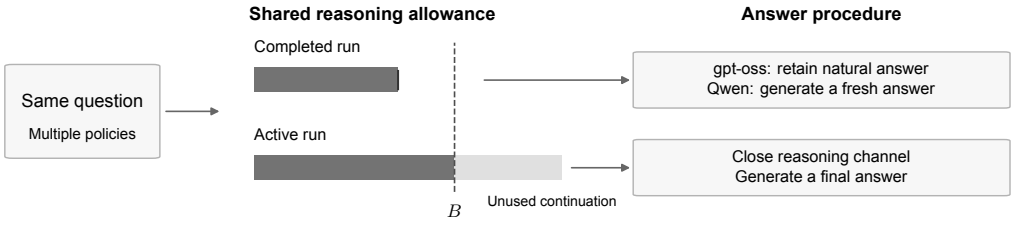


Figure 2: **The same reasoning allowance supports distinct behavioral and diagnostic questions.** The illustrative traces show a completed and an active run evaluated at a shared B. Active runs receive a generated answer after channel closure. Completed gpt-oss runs retain their natural answer, whereas the Qwen prompt continuations generate a fresh answer from the complete reasoning block. Generated answers have allowance $R = 64$ for Qwen and $R = 16$ for gpt-oss, and every item is retained. Completion-state comparisons and common-context prefix readouts diagnose the resulting differences. The diagnostic replay comparisons are in Figure A2 and Table A8.

et al. (2026) use forced answers to study intermediate predictions and answer stabilization, and McFadyen et al. (2026) argue for evaluation across shared inference-compute ranges. Building on these approaches, we examine how a reasoning control changes early answer behavior and how completion-state selection qualifies its interpretation (Table A1). Our contribution is to separate these claims empirically, rather than to report fixed budgets or changing rankings alone. Budget forcing in s1 ends the reasoning block at a token limit and elicits the answer (Muennighoff et al., 2025), as does the documented Qwen3 thinking budget (Qwen Team, 2025), and our cap-then-answer evaluation applies this procedure to every policy at shared allowances. L1 trains a model to follow a length stated in the prompt (Aggarwal & Welleck, 2025), whereas we test whether an untrained instruction is followed and separate finishing earlier from adapting to the stated number.

Early-stopping methods seek to avoid reasoning that adds little to the answer. Liu & Wang (2025) use answer convergence as a stopping signal, DEER tests confidence in trial answers at reasoning transitions (Yang et al., 2026), and REFRAIN detects redundant reflection and adapts its stopping threshold (Sun et al., 2025a). Allocating compute by question is a complementary route: Snell et al. (2024) show that effective test-time compute allocation depends on prompt difficulty. Both lines motivate separating final accuracy and cost from performance under shared allowances.

Forced intermediate answers (Datta et al., 2026) and prefix replay with answer-token readouts (Ballon et al., 2026b) both read answers off partial reasoning, and we use the latter readout for the candidate-logit diagnostics. Relative to these diagnostics, we add paired interventions on reasoning length and effort at shared allowances, generated-answer continuations that check the candidate-logit readout behaviorally, and a completion-state decomposition of the resulting differences.

3 Shared allowances and answer readouts

Let B be an externally chosen reasoning allowance, z a generation policy, and L^z its terminal reasoning length. We compare every policy on the same questions and allowances (Figure 2). Natural completion measures final accuracy and realized generated tokens. Fixed-allowance evaluation measures the answer obtained after at most B reasoning tokens, with a reserved final-answer allowance R. The behavioral contrast is

$$\Delta_{z,z'}(B) = \mathbb{E}[Y_z^{\mathrm{cap}}(B) - Y_{z'}^{\mathrm{cap}}(B)] \,.$$

The answer procedure is held fixed within each comparison. In the gpt-oss cap-then-answer evaluation, a completed run retains its natural answer. An active run is interrupted at B, its reasoning channel is closed, and a greedy final answer is generated in its source effort context. The Qwen prompt experiment instead generates a fresh answer from each capped reasoning block, including blocks that have completed, in the source instruction context. We report that continuation endpoint separately from terminal-aware candidate-logit scoring, which retains natural answers after completion.

The main three-replicate cohorts contain all 198 GPQA Diamond items and the same 500 MMLU-Pro items. gpt-oss low, medium and high effort are evaluated at $B \in \{128, 256, 512, 1{,}024, 2{,}048, 4{,}096, 8{,}192\}$. The nine-instruction Qwen prompt experiment (the prompt-selection cohort) uses eight generations per item on MMLU-Pro, and a replication repeats its six generated-answer instructions on 500 further MMLU-Pro items not used to choose the prompts, with the procedure fixed in advance (Appendix A.14). Its generated-answer comparisons cover six instructions at $B \in \{512, 2{,}048\}$, while the broader ordinary-versus-numeric/concision curves use candidate logits at $B \in \{512, 2{,}048, 8{,}192, 16{,}384\}$.

The concise instruction is: "Use a very concise reasoning strategy. Prioritize reaching a usable answer quickly." It states no number. The numeric/concision instruction instead announces an allowance inside the reasoning block, requests a strategy appropriate for that budget, and asks the model to close the block when it has enough evidence. The advance exact-stop instruction only announces that reasoning will be stopped after the stated number of tokens. We refer to these two numeric instructions together as budget instructions. Crossing the two announcements with both actual horizons separates generic instruction effects from any extra benefit of matching the number (verbatim prompts in Table A9). gpt-oss effort is set through the checkpoint's low, medium or high chat-template setting.

Multiple-choice continuations use greedy decoding with $R = 64$ for the six-instruction Qwen panel and $R = 16$ for gpt-oss. Missing or malformed answers count as incorrect. The answer stage is additional to B, so reasoning allowances are neither total-output limits nor measured wall-clock deadlines. We report arithmetic-mean generated tokens including the answer, with medians and 90th percentiles. Paired bootstrap intervals resample items while retaining each item's outcomes across all policies and horizons. The three-replicate analyses average replicates within item. The eight-replicate prompt study also resamples replicates within item. Captions distinguish pointwise intervals from simultaneous bands.

Diagnostics ask what answer is obtainable under a specified readout. Common-context replay places each stored prefix in the ordinary Qwen or medium-effort gpt-oss prompt and applies a shared closure. The candidate-logit readout selects among valid answer-token spellings and normalizes probabilities over options. Replayed prefixes exclude original final-answer tokens. A jointly unfinished comparison further conditions on both runs remaining active, so it describes a selected population. Generated continuations and question-specific prefix controls test the behavior of these readouts (Figure 5). They do not identify when a model decided.

4 Effort changes the accuracy–allowance comparison

Low effort has the highest observed accuracy at the three shortest allowances in all four model–benchmark comparisons (Figure 1). At $B = 512$, its advantage over high effort ranges from $+4.5$ to $+18.9$ points, with all four simultaneous intervals above zero. By $B = 8{,}192$, high effort is 4.4–8.8 points more accurate than low. All four pointwise intervals exclude zero, while the simultaneous interval includes zero for gpt-oss-20b on GPQA Diamond (Table A4). These curves compare policies on the evaluation cohorts, without testing a policy chosen on separate questions.

The early generated-answer advantage is concentrated in pairs where lower effort has completed and high effort remains active (Figure 3). At $B = 512$, this stratum contributes $+3.3$ to $+16.3$ points and is the largest positive component in all eight lower-versus-high comparisons. Re-pairing the three high-effort replicates within each item 10,000 times leaves it the largest contributor in every draw (Table A6). This analysis locates the observed advantage. It does not establish that changing stopping alone, while holding reasoning fixed, would reproduce it.

Common-context candidate-logit readouts show the same completion concentration, with smaller, mixed differences among jointly unfinished pairs (Figure 3e,f). None of those item-weighted conditional intervals excludes zero, but their precision varies substantially, and the pair-weighted contrast for gpt-oss-120b on GPQA Diamond, $+14.4$ ($[+3.8, +25.2]$), does exclude it (Table A25). For gpt-oss-120b on GPQA Diamond, the low-minus-high estimate is $+10.6$ points with an interval of $[-1.6, +22.8]$, based on only 132 jointly unfinished pairs from 63 items. This leaves room for a substantial positive effect in that selected group. For gpt-oss-20b on MMLU-Pro, the medium-minus-high estimate is -1.6 points with an interval of $[-5.7, +2.6]$ (Table A24).

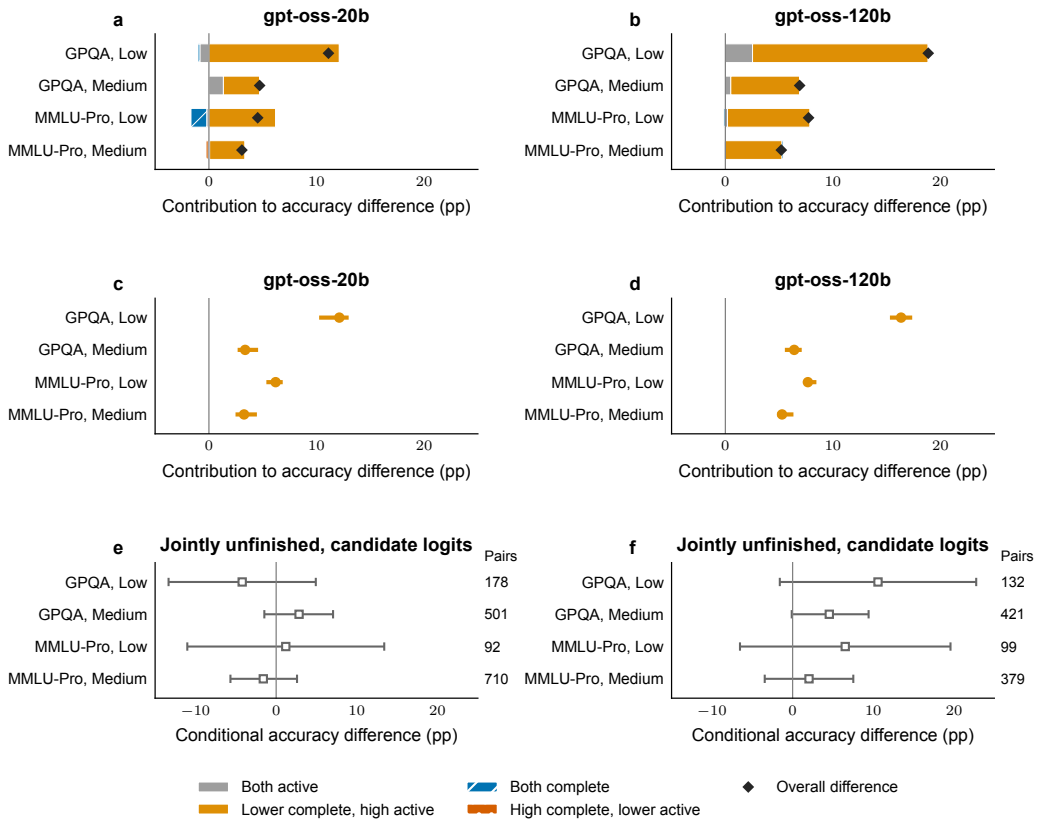


Figure 3: **The early advantage concentrates in completed runs, with uncertain differences among unfinished prefixes.** Columns show gpt-oss-20b and -120b at $B = 512$ on 198 GPQA Diamond and 500 MMLU-Pro items, with three generations per effort. **a,b**, Full-cohort generated-answer contributions of the four completion strata sum to the overall difference (diamond). Completed runs retain their natural answer and active runs receive a source-policy continuation with $R = 16$. **c,d**, Lower-complete/high-active contribution under the original pairing and its full range across 10,000 within-item re-pairings. These are sensitivity ranges, not sampling intervals. **e,f**, Common medium-effort candidate-logit accuracy differences among jointly unfinished pairs, with pair counts and pointwise 95% item-bootstrap intervals. These selected-population intervals do not establish equivalence (Table A24).

Conditioning on both runs remaining active also changes which questions contribute. If a policy finishes easier questions earlier, those questions disappear from its jointly unfinished comparison. A small conditional gap therefore need not imply a small effect in the original population. The agreement between the full-cohort curve and the selected-prefix contrast on where the gap concentrates supports the descriptive conclusion without identifying a stopping-only mechanism.

Evaluating high effort at the amount of reasoning lower effort elects to use gives a complementary view. Completed low- or medium-effort reasoning exceeds the matched-horizon high-effort prefix by 14.5–26.3 points under common medium-effort candidate-logit scoring, and by 15.5–25.9 points under generated continuations, with intervals above zero throughout (Figure 5b and Table A22). Because lower effort selects the horizon, this result includes its stopping decision.

Earlier completion can be valuable even when high effort eventually gives a better answer. On GPQA Diamond, low gpt-oss-120b effort reaches 62.3% accuracy at 512 tokens, versus 43.4% for high effort. At natural completion, high effort reaches 79.3%, versus 65.0% for low, but uses an arithmetic mean of 11,291 generated tokens rather than 339. Its median is 4,402 and its 90th percentile is 33,107, illustrating why typical length alone does not measure expected token expenditure (Table 1).

Table 1: **Natural accuracy improves with effort at substantially higher expected token expenditure, while the concise instruction reduces Qwen3-14B's expenditure by about a third with an estimated one-point accuracy loss in the replication.** The full 198-item GPQA Diamond and 500-item MMLU-Pro gpt-oss cohorts use three generations per effort. The Qwen3-14B rows use the prompt-selection cohort and the replication cohort, each 500 MMLU-Pro items with eight generations per instruction, and count reasoning and final-answer tokens. The replication's paired natural-accuracy difference is -1.0 points ($[-2.3, +0.2]$, Table A39d). Token statistics include reasoning, channel boundaries and final answers. Runs that reached the generation limit remain included; missing or malformed final answers count as incorrect. Values are observed summaries without uncertainty intervals. Parsing counts are in Table A5.

Model / benchmark	Effort or instruction	Accuracy (%)	Mean tokens	Median	90th percentile
gpt-oss-20b / GPQA	Low	55.9	498	294	1,137
	Medium	64.6	9,167	1,816	29,555
	High	66.0	37,120	14,468	119,114
gpt-oss-20b / MMLU-Pro	Low	64.2	171	100	390
	Medium	72.3	1,116	496	2,400
	High	74.5	7,792	1,414	23,087
gpt-oss-120b / GPQA	Low	65.0	339	264	689
	Medium	68.5	1,466	934	3,618
	High	79.3	11,291	4,402	33,107
gpt-oss-120b / MMLU-Pro	Low	75.5	171	89	409
	Medium	79.1	517	266	1,162
	High	79.8	3,669	972	8,964
Qwen3-14B / MMLU-Pro (prompt-selection cohort)	Ordinary	75.5	2,117	1,090	5,259
	Concise	75.6	1,469	683	3,408
Qwen3-14B / MMLU-Pro (replication)	Ordinary	76.6	2,358	1,124	5,837
	Concise	75.6	1,626	726	3,873

5 Concision and announced-budget matching have different effects

The concise instruction improves Qwen's early generated answers (Figure 4a,b). In the replication on 500 questions not used to choose the prompts, accuracy rises from 59.6% to 63.1% at $B = 512$ ($+3.5$ points, 95% CI $[+1.8, +5.2]$) and from 70.7% to 73.2% at $B = 2{,}048$ ($+2.5$, $[+0.9, +4.1]$; Table A39). In the prompt-selection cohort shown in Figure 4, the gains are $+3.9$ ($[+2.1, +5.7]$) and $+2.1$ ($[+0.6, +3.6]$) points, median reasoning length falls from 1,083 to 679 tokens (Figure 4d), and at natural completion accuracy is 75.5% versus 75.6% while mean generated tokens fall from 2,117 to 1,469 (Table 1). In the prompt-selection cohort, pairs where only one answer parses contribute 1.4 of the 2.1-point gain at $B = 2{,}048$, versus 0.6 of 3.9 points at $B = 512$ (Appendix A.2), and among pairs whose two answers both parse the concise advantage is $+3.5$ points at 512 tokens ($[+1.7, +5.3]$) and $+0.7$ at 2,048 ($[-0.7, +2.1]$), so the later gain mostly comes from the concise instruction producing more parseable answers (Table A38).

Matching the announced number to the actual allowance gives much smaller estimated benefits (Figure 4c). The generated-answer readout gives -0.1 points for numeric/concision prompting ($[-0.8, +0.7]$) and $+0.2$ for advance exact-stop notice ($[-0.5, +0.9]$, Table A17). The candidate-logit contrasts are -0.2 ($[-1.0, +0.6]$) and $+0.1$ ($[-0.6, +0.8]$), respectively. These intervals leave room for small benefits and do not establish equivalence.

Shorter traces alone give a different impression of the numeric/concision instruction (Figure A13a,b,f). It reduces typical Qwen3-14B reasoning length by 12–17%, while same-horizon candidate-logit accuracy differences range from -0.5 to $+1.3$ points on GPQA Diamond and from -0.9 to $+2.0$ on MMLU-Pro, with only the MMLU-Pro 512-token contrast simultaneously above zero. Median length changes little as the announced allowance increases 32-fold, with within-item log–log slopes of 0.012 and 0.018 (Table A18). Thus, the instruction changes typical length more clearly than it targets the announced count. Completion timing shows the same pattern. Every numeric instruction raises natural completion before the stop by 2.6–6.3 points, whereas stating the actual horizon rather than the other number adds only 0.5–0.7 points with pointwise 95% intervals including zero, against 16.1 and 10.1 points for the concise instruction (Table A37). Consistent with this, hidden-state readouts recover the exact announced budget at the prompt boundary but mostly fall to label-shuffle levels

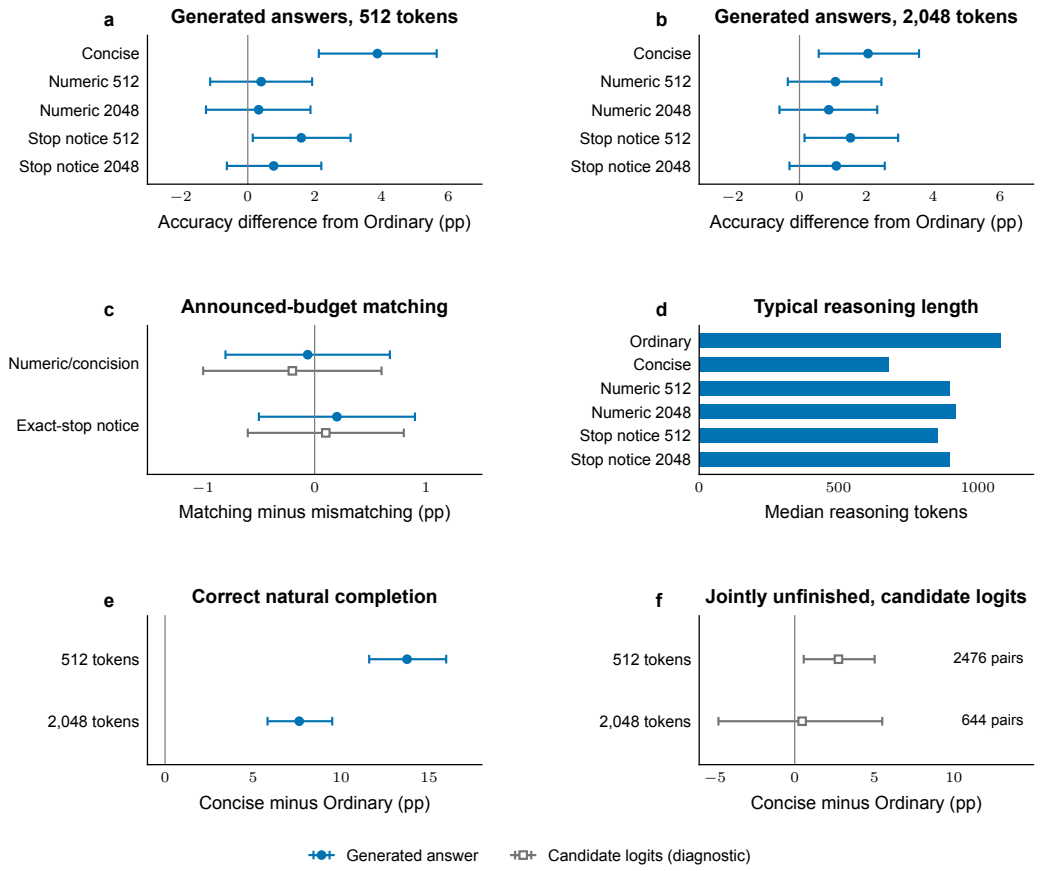


Figure 4: **Concision improves early generated answers, while matching the announced number adds little.** Qwen3-14B in the prompt-selection cohort, 500 MMLU-Pro items with eight paired generations per instruction. The replication cohort's contrasts are in Table A39. **a,b**, Generated-answer accuracy differences from ordinary generation at 512 and 2,048 reasoning tokens. Every capped reasoning block, including completed blocks, receives a fresh source-policy answer with $R = 64$. Parsing failures are incorrect. **c**, Matching-number contrasts under generated-answer and candidate-logit scoring. **d**, Observed median reasoning lengths. **e**, Concise-minus-ordinary differences in correct natural completion by each horizon. **f**, Common-context candidate-logit differences among jointly unfinished pairs, with the selected pair counts. Error bars are pointwise 95% nested item–replicate bootstrap intervals for each panel's endpoint. Candidate-logit intervals do not describe generated-answer uncertainty (Tables A15 and A17).

once reasoning begins, and text classifiers cannot separate budget-prompted from ordinary traces by 512 reasoning tokens (Table A36 and Figure A11).

The concise instruction also changes completion and selected unfinished-prefix outcomes (Figure 4e,f). Correct natural completion by 512 tokens rises from 18.2% to 31.9% (+13.8 points, [+11.6, +16.0]). By 2,048, it rises from 58.0% to 65.6% (+7.6, [+5.8, +9.5]). Under common-context candidate-logit replay, the 512-token all-pair advantage is +3.1 points ([+1.5, +4.8]), and the difference among 2,476 jointly unfinished pairs is +2.7 ([+0.6, +5.0], Table A15). At 2,048, only 644 pairs remain jointly unfinished, and the estimate is +0.5 ([−4.8, +5.5]). This contrast gives positive evidence within the early selected group, with less precision later, without establishing a population-wide improvement in unfinished reasoning.

Prompt selection and the answer-allowance check used the cohort shown in Figure 4, so those comparisons do not test the prompts on new questions. The replication also gives matching contrasts close to zero, +0.3 and −0.2 points with intervals including zero, and among pairs that both policies leave

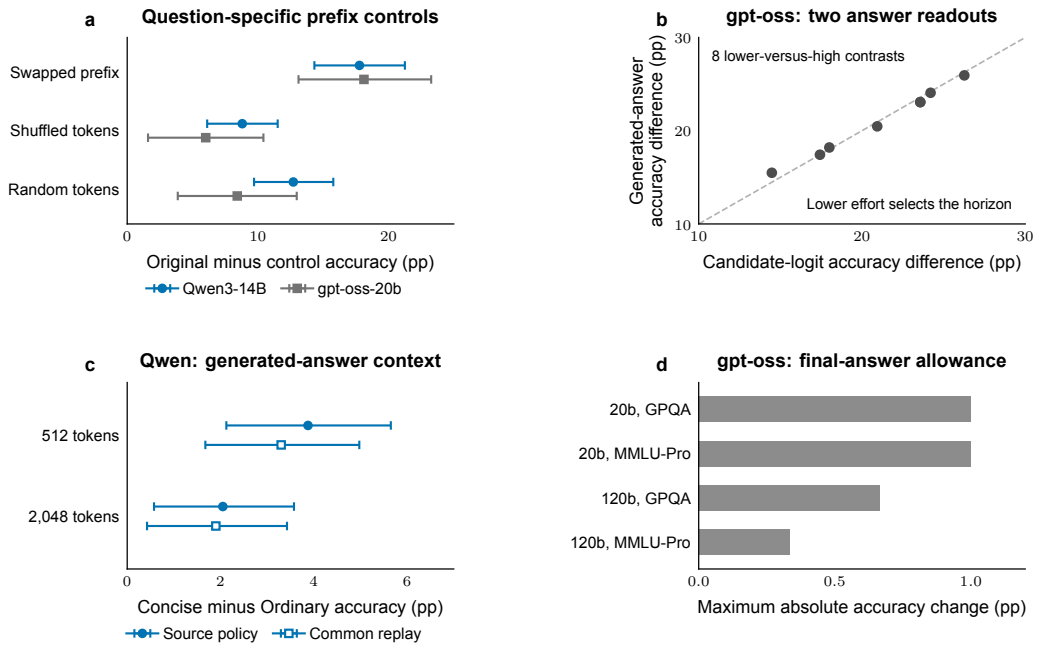


Figure 5: **The readout checks support the observed contrasts across specified answer procedures.** **a**, Original-minus-control candidate-logit accuracy on matched GPQA/MMLU-Pro calibration cohorts: 511 item–horizon pairs from 366 items for gpt-oss-20b and 963 from 599 for Qwen. Error bars are pointwise 95% paired item-bootstrap intervals. **b**, Eight gpt-oss lower-versus-high contrasts at the lower-effort policy's terminal horizon, under common medium-effort candidate-logit and generated-answer readouts. Points are observed estimates and the dashed line denotes equality. The cohorts with reconstructable paired prefixes contain 198 GPQA or 500 MMLU-Pro items and three generations. Already-completed high-effort reasoning is replayed. **c**, Qwen concise-minus-ordinary generated-answer effects in the prompt-selection cohort, in source-policy and common contexts on 500 items with eight replicates. Every reasoning block receives a fresh $R = 64$ answer, including completed blocks. Error bars are pointwise 95% nested item–replicate bootstrap intervals. **d**, Maximum absolute $R = 16$ versus $R = 64$ accuracy change across all efforts and seven horizons, on the same 100 items per benchmark with three replicates. Completed answers are retained. Bars are observed maxima, including replay-batching sensitivity, not confidence intervals.

unfinished at 512 tokens the concise advantage in generated answers is $+4.1$ points ($[+1.7, +6.4]$) in the source-policy context and $+4.7$ ($[+2.6, +6.9]$) in the common replay context, positive under every within-item re-pairing (Tables A39 and A40). With completed blocks keeping their natural answers, the gains are 3.5 and 1.8 points (Table A39c). At natural completion, the replication's concise accuracy is 75.6% against 76.6% for ordinary generation (-1.0 points, $[-2.3, +0.2]$) with mean tokens down from 2,358 to 1,626 (Table A39d and Table 1). A second MMLU-Pro subset agrees (Table A33).

6 Readout checks and open-ended scope

Question-specific prefixes improve the diagnostic answer readout (Figure 5a). Replacing a prefix with another question's length-matched prefix reduces accuracy by 18.1 points for gpt-oss-20b and 17.8 for Qwen3-14B. Terminal replay agrees with parsed natural answers in 98.3–100.0% of a separate validation cohort (Table A11). These checks show that the readout uses information specific to the preceding question and reasoning, not an intervention-free belief or a time at which the model had already decided.

Generated answers provide a behavioral check beyond candidate probabilities. The gpt-oss policy-selected-horizon effects agree closely across those readouts (Figure 5b), and Qwen's concise advan-

tage remains positive when the final answer is generated in a common context (Figure 5c). Increasing the gpt-oss final-answer allowance from $R = 16$ to $R = 64$ changes accuracy by at most one point in any condition on the tested subset (Figure 5d and Table A7). These checks support the reported comparisons under the tested finalization procedures.

The open-ended Omni-MATH-2 comparisons test whether the pattern depends on selecting from listed options (Table A12). GPT-5-mini judges the generated free-form answers, using the grading procedure and prior human audit of judge disagreements reported by Ballon et al. (2026a). At lower effort's chosen terminal horizon, gpt-oss's generated-answer advantage over high effort is 33.5–42.0 points, with all item-bootstrap intervals above zero, on 475 available paired items for gpt-oss-20b and 490 for -120b. This extends the policy-selected-horizon result to open-ended answers. Qwen's numeric/concision effects are mixed: $+3.5$, $+2.0$ and -4.1 points at 512, $2{,}048$ and $8{,}192$ tokens, with all intervals including zero. These comparisons retain only the 170, 150 and 73 jointly judged items at each horizon from a 200-item sample. Both model families use one rollout per condition here, so the extension provides limited evidence on task scope and no estimate of variation across independent generations.

7 Discussion and Conclusion

Shortening reasoning, delivering a useful early answer and adapting to an announced budget are distinct achievements. The gpt-oss results show that lower effort can improve answers under short allowances while losing its advantage as the allowance grows. The early difference concentrates in runs that lower effort has already completed. Earlier completion is useful when it delivers a correct answer sooner, but it need not imply uniformly better unfinished reasoning. For Qwen, the concise instruction improves early answers while matching the announced number has much smaller estimated effects, and a replication on questions not used to choose the prompts reproduces both.

Both interfaces change when the model finishes. For Qwen, matching the announced number adds little to accuracy or completion, with intervals that include zero (Figure 4c and Table A37), adding that number to the concise instruction does not consistently improve on it (Table A41), and for gpt-oss, adding budget wording to an effort setting gives no broad gain at 512 tokens (Table A42). The effort setting is also more visible in the reasoning itself. Effort is classifiable from the first 64 reasoning tokens at 0.70 against a 0.33 chance level, whereas the exact announced budget mostly falls to shuffle level in Qwen's hidden states once reasoning begins and the budget-prompt text signal decays to chance by 512 tokens, although a coarser binary prompt signal remains modestly decodable (Figure A11 and Table A36). These are descriptive contrasts between two case studies, because interface is confounded with model family and training.

Evaluations should therefore lead with natural accuracy and realized token cost, then compare generated answers across shared allowances using an explicit finalization procedure. The appropriate reasoning interface depends on the required accuracy and allowance, rather than an accuracy-per-token ratio. Shorter traces do not by themselves establish better early answers, and early-answer gains do not by themselves establish uniformly better unfinished reasoning.

8 Limitations

The main evidence covers two model families and two multiple-choice benchmarks, using different prompt and effort interventions. It does not establish a causal comparison between those interfaces or universal benefits of concision. Three generations per policy limit what the main cohorts can say about sampling variation, while the prompt study uses eight. Cohorts reconstructed from text or using one rollout provide more limited replication (Table A3). The Qwen replication covers one model, one benchmark and the six generated-answer instructions, and its unfinished-pair contrast conditions on the 64% of pairs still active at 512 tokens. The open-ended Qwen comparison tests the numeric/concision instruction rather than the concise instruction, and the concise instruction bundles a request for concise reasoning with a request to reach a usable answer quickly, which the experiments do not separate. Generated answers depend on channel closure, answer allowance and readout context, with the Qwen continuation and gpt-oss retention rules stated separately. Token-budget replay does not measure online latency.

AI USE STATEMENT

Generative AI assistants (OpenAI Codex/ChatGPT and Anthropic Claude) assisted with literature retrieval, language revision and section drafting, manuscript organization, feedback on the research questions, experimental design and analysis plans, implementation of experimental, data-processing, analysis and figure code, interpretation of results, and consistency checks. Language models also generated the reasoning traces and answers described in the experimental procedures, and GPT-5-mini judged the open-ended mathematics comparisons and annotated reasoning excerpts for the structural-marker analysis, with the earlier human validation of mathematics judging described in the appendix. The authors reviewed and verified the resulting text, code, references, analyses and claims, and take responsibility for the final content of this work.

ETHICS STATEMENT

This study uses public benchmarks and open-weight models and involves no human subjects or personal data. The reasoning traces were generated for evaluation only, and the released outcome records contain no trace text, only token and configuration hashes. Controls that shorten reasoning lower cost and latency but also remove reasoning that supports auditing. Our results concern how such controls should be evaluated, not whether they should be deployed.

REPRODUCIBILITY STATEMENT

The code and data for this study will be released in public repositories on acceptance. The anonymous supplement contains the analysis code and the outcome records. The GPQA Diamond dataset (Rein et al., 2024) is available at `https://huggingface.co/datasets/Idavidrein/gpqa`, MMLU-Pro (Wang et al., 2024) at `https://huggingface.co/datasets/TIGER-Lab/MMLU-Pro`, and Omni-MATH-2 (Ballon et al., 2026a) at `https://huggingface.co/datasets/martheballon/Omni-MATH-2`. The Qwen3 model family (Yang et al., 2025) is available at `https://huggingface.co/collections/Qwen/qwen3` and the gpt-oss family (Agarwal et al., 2025) at `https://huggingface.co/collections/openai/gpt-oss`. Software versions and environments are listed in Appendix A.6.

REFERENCES

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025. URL https://arxiv.org/abs/2508.10925.

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. *arXiv preprint arXiv:2503.04697*, 2025. URL https://arxiv.org/abs/2503.04697.

Pranjal Aggarwal, Marjan Ghazvininejad, Seungone Kim, Ilia Kulikov, Jack Lanchantin, Xian Li, Tianjian Li, Bo Liu, Graham Neubig, Anaelia Ovalle, et al. Reasoning over mathematical objects: on-policy reward modeling and test time aggregation. *arXiv preprint arXiv:2603.18886*, 2026. URL https://arxiv.org/abs/2603.18886.

Daman Arora and Andrea Zanette. Training language models to reason efficiently. *arXiv preprint arXiv:2502.04463*, 2025. URL https://arxiv.org/abs/2502.04463.

Marthe Ballon, Andres Algaba, and Vincent Ginis. The relationship between reasoning and performance in large language models—o3 (mini) Thinks Harder, Not Longer. *arXiv preprint arXiv:2502.15631*, 2025. URL https://arxiv.org/abs/2502.15631.

Marthe Ballon, Andres Algaba, Brecht Verbeken, and Vincent Ginis. Benchmarks saturate when the model gets smarter than the judge. *arXiv preprint arXiv:2601.19532*, 2026a. URL https://arxiv.org/abs/2601.19532.

Marthe Ballon, Brecht Verbeken, Vincent Ginis, and Andres Algaba. Probing the trajectories of reasoning traces in large language models. *arXiv preprint arXiv:2601.23163*, 2026b. URL https://arxiv.org/abs/2601.23163.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for 2+3=? On the overthinking of o1-like LLMs. *arXiv preprint arXiv:2412.21187*, 2024. URL https://arxiv.org/abs/2412.21187.

Ayan Datta, Zhixue Zhao, Bhuvanesh Verma, Radhika Mamidi, Mounika Marreddy, and Alexander Mehler. Large language models decide early and explain later. *arXiv preprint arXiv:2604.22266*, 2026. URL https://arxiv.org/abs/2604.22266.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, et al. Omni-MATH: A universal olympiad level mathematic benchmark for large language models. *arXiv preprint arXiv:2410.07985*, 2024. URL https://arxiv.org/abs/2410.07985.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL https://doi.org/10.1038/s41586-025-09422-z.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24842–24855. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.1274. URL https://aclanthology.org/2025.findings-acl.1274/.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*, 2024. URL https://arxiv.org/abs/2412.16720.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022. doi: 10.52202/068431-1613. URL https://doi.org/10.52202/068431-1613.

Junyan Li, Wenshuo Zhao, Yang Zhang, and Chuang Gan. Steering LLM thinking with budget guidance. *arXiv preprint arXiv:2506.13752*, 2025. URL https://arxiv.org/abs/2506.13752.

Xin Liu and Lu Wang. Answer convergence as a signal for early stopping in reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17896–17907. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.904. URL https://aclanthology.org/2025.emnlp-main.904/.

Jessica McFadyen, Ole Jorgensen, Harry Coppock, Kevin Wei, and Cozmin Ududec. How inference compute shapes frontier LLM evaluation. *arXiv preprint arXiv:2606.17930*, 2026. URL https://arxiv.org/abs/2606.17930.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947. doi: 10.1007/BF02295996. URL https://doi.org/10.1007/BF02295996.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20286–20332. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1025. URL https://aclanthology.org/2025.emnlp-main.1025/.

Sania Nayab, Giulio Rossolini, Marco Simoni, Andrea Saracino, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. Concise thoughts: Impact of output length on LLM reasoning and cost. *arXiv preprint arXiv:2407.19825*, 2024. URL https://arxiv.org/abs/2407.19825.

Qwen Team. Qwen3 thinking budget. https://github.com/QwenLM/Qwen3/blob/main/docs/source/getting_started/thinking_budget.md, 2025. Accessed 20 June 2026.

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

Matthew Renze and Erhan Guven. The benefits of a concise chain of thought on problem-solving in large language models. *arXiv preprint arXiv:2401.05618*, 2024. URL https://arxiv.org/abs/2401.05618.

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

Jinyan Su, Jennifer Healey, Preslav Nakov, and Claire Cardie. Between underthinking and overthinking: An empirical study of reasoning length and correctness in LLMs. *arXiv preprint arXiv:2505.00127*, 2025. URL https://arxiv.org/abs/2505.00127.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, et al. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025. URL https://arxiv.org/abs/2503.16419.

Renliang Sun, Wei Cheng, Dawei Li, Haifeng Chen, and Wei Wang. Stop when enough: Adaptive early-stopping for chain-of-thought reasoning. *arXiv preprint arXiv:2510.10103*, 2025a. URL https://arxiv.org/abs/2510.10103.

Yi Sun, Han Wang, Jiaqiang Li, Jiacheng Liu, Xiangyu Li, Hao Wen, Yizhen Yuan, Huiwen Zheng, Yan Liang, Yuanchun Li, and Yunxin Liu. An empirical study of LLM reasoning ability under strict output length constraint. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7652–7671. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.emnlp-main.389. URL https://aclanthology.org/2025.emnlp-main.389/.

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. *arXiv preprint arXiv:2203.11171*, 2022. URL https://arxiv.org/abs/2203.11171.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *Advances in Neural Information Processing Systems*, 37:95266–95290, 2024. doi: 10.52202/079017-3018. URL https://doi.org/10.52202/079017-3018.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. doi: 10.52202/068431-1800. URL https://doi.org/10.52202/068431-1800.

Hao Wen, Xinrui Wu, Yi Sun, Feifei Zhang, Liye Chen, Jie Wang, Yunxin Liu, Yunhao Liu, Ya-Qin Zhang, and Yuanchun Li. BudgetThinker: Empowering budget-aware LLM reasoning with control tokens. *arXiv preprint arXiv:2508.17196*, 2025. URL https://arxiv.org/abs/2508.17196.

Tsung-Han Wu, Mihran Miroyan, David M. Chan, Trevor Darrell, Narges Norouzi, and Joseph E. Gonzalez. Are large reasoning models interruptible? *arXiv preprint arXiv:2510.11713*, 2025. URL https://arxiv.org/abs/2510.11713.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*, 2025. URL https://arxiv.org/abs/2502.18600.

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

Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Qiaowei Li, Minghui Chen, Zheng Lin, and Weiping Wang. Dynamic early exit in reasoning models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8df90a1440ce782d1f5607b7a38f2531-Abstract-Conference.html.

A Appendix

A.1 Unified generated-answer evaluation

The seven-horizon evaluation uses 6,282 independently generated source trajectories per gpt-oss model: three efforts, three replicates, 198 GPQA Diamond items, and 500 MMLU-Pro items. It produces 43,974 outcomes per model without outcome-based exclusions. The source generations use the model checkpoints and settings specified below. Answer continuations use vLLM 0.21.0, PyTorch 2.11.0, and Transformers 5.9.0, with greedy decoding, an explicit 16-token answer allowance, and a shared policy-specific final-channel closure. Natural final answers all fit within that allowance. We record generation limits, parsing failures and channel boundaries. Every unavailable or unparseable answer is incorrect.

Stored completion IDs determine the reasoning boundary and prefix exactly. Original generation-time prompt IDs were not stored for the gpt-oss generations. We reconstruct the question, system instruction, and source effort with the original checkpoint's chat template, fixing its otherwise dynamic date field to 2026-09-12. Reconstructed prompt-token counts agree with every stored count; this does not establish identical original prompt bytes or date metadata. The evaluation uses exact reasoning prefixes and source-policy instructions without recovering the original generation-time cache. Source-record, completion-token, reconstructed-prompt, prefix, template, and configuration hashes accompany each outcome.

The prefix checks cover 12,564 source trajectories and all 87,948 evaluations. No replay prefix includes original final-channel tokens, all requested prefixes are available, and all realized generated-token counts are at most $B + R$. There are 53 sources without a natural final boundary for gpt-oss-20b and one for -120b; each still supplies the required unfinished prefixes. Across all policies and horizons, the generated answers contain 150 parsing failures for gpt-oss-20b and 36 for -120b, all counted as incorrect. Natural generated boundary tokens are included in cost; externally supplied closure tokens are input overhead, not newly generated output. These offline replay costs do not measure online interruption latency.

Accuracy, correct natural completion, completion probability, and arithmetic-mean generated tokens are computed on the same full cohort at each horizon. Median and 90th-percentile token counts are also reported. The paired bootstrap averages the three replicates within item and draws 10,000 item samples, preserving each complete policy–horizon vector. Pointwise intervals are percentile intervals; a simultaneous band uses the 95th percentile of the largest absolute centered bootstrap difference across both lower-effort contrasts and all seven horizons within each model–benchmark comparison. Replicate re-pairing independently permutes the three high-effort outcomes within item in each draw, preserving each balanced unconditional mean. Its ranges are distinct from the sampling intervals.

The final-answer allowance comparison uses 100 items per benchmark with seed 20260912, retaining all efforts, replicates, and horizons. It re-evaluates the same prefixes with $R = 64$ and compares them to their $R = 16$ outcomes (Table A7). Because the serving batches differ, this is a combined allowance and numerical reproducibility check, not an isolated causal estimate of increasing R. We use no equivalence margin for these reused cohorts.

A.2 Generated-answer matching in the Qwen prompt experiment

The generated-answer prompt panel covers six of the nine instructions, two horizons, two answer contexts, 500 items and eight generations, giving 96,000 outcomes. We generate fresh greedy answers with $R = 64$ from the exact stored prompt, reasoning-prefix and closure token IDs, including completed reasoning blocks. The continuations use vLLM 0.21.0, Transformers 5.9.0 and PyTorch 2.11.0. All prefixes pass boundary and context-length checks. The specified parser counts all 2,580 unparseable answers as incorrect. Each context is analyzed separately.

For numeric/concision and advance exact-stop instructions, the matching contrast is half the sum of the $H = 512$ minus $H = 2{,}048$ accuracy difference at $B = 512$ and the reverse difference at $B = 2{,}048$. The 10,000-draw nested bootstrap first samples items and then replicates within each sampled item, preserving the entire instruction–horizon vector. We report pointwise percentile intervals without an equivalence margin. The answer allowance was selected using an ordinary/concise sensitivity check

on the same cohort. The generated-answer and candidate-logit comparisons share the selected items and reasoning trajectories, so they are not independent replications (Table A17).

Independent checks reproduce all decoded answers and parser outcomes. Of the 2,580 full-answer parse failures, 2,560 exhaust the answer allowance. An AI-assisted inspection of 129 outputs, including all naturally stopped failures, finds small format-rule limitations that do not change the source-policy concise contrasts. This is not human grading, and parseability is part of the endpoint. At $B = 2{,}048$, pairs where only one answer parses contribute 1.4 of the 2.1 percentage-point concise advantage. At $B = 512$, they contribute 0.60 of 3.875 points. These are descriptive contributions, not effects of changing the parser or reasoning process.

A.3 Candidate-logit estimands and replay rules

For Qwen3-14B, let h denote ordinary generation with the horizon hidden (the *surprise stop*) and v the textual numeric/concision treatment with the horizon visible (the *bundled prompt*). The terminal-aware policy-context readout uses $Y_{ir}^{z,T_{\mathrm{pol}}}(B) = Y_{ir}^{z,N}(L_{ir}^{z})$ when the trace finishes before B, and $Y_{ir}^{z,P_{\mathrm{pol}}}(B)$ otherwise. The candidate-logit contrast is

$$\tau_{Q,\mathrm{pol}}(B) = \mathbb{E}_{i,r}\left[Y_{ir}^{v,T_{\mathrm{pol}}}(B) - Y_{ir}^{h,T_{\mathrm{pol}}}(B)\right].$$

Both arms therefore face the same imposed horizon, although a naturally completed trace can contain fewer than B tokens. For any tested instruction a, we also report correct natural completion by the deadline,

$$K_{ir}^{a}(B) = \mathbf{1}\{L_{ir}^{a} \le B\}\, Y_{ir}^{a,N},$$

and the both-active common-context contrast

$$\tau_{Q,\mathrm{BA}}^{a}(B) = \mathbb{E}_{i,r}\Big[Y_{ir}^{a,P_{\mathrm{com}}}(B) - Y_{ir}^{h,P_{\mathrm{com}}}(B) \mid L_{ir}^{a} > B,\ L_{ir}^{h} > B\Big].$$

The four-stratum decomposition partitions every pair into both active, instruction complete/ordinary active, ordinary complete/instruction active, and both complete. It reports each stratum's frequency, within-stratum contrast, and contribution to the unconditional terminal-aware difference. Within-stratum contrasts average replicate outcomes within item and then across items, whereas shares and contributions use raw pair means, so a within-stratum contrast and its contribution can differ in sign when items contribute unequal numbers of pairs to a stratum (Table A24). Table A25 reports both conventions with item-cluster intervals for the jointly unfinished stratum. The literal cap-then-answer experiment and the continuation check of the instruction arms replace candidate-logit scoring with the behavioral readout C at $B \in \{512, 2{,}048\}$.

For gpt-oss, let $c \in \{\mathrm{low}, \mathrm{medium}\}$ denote the lower effort. The matched horizon is the lower-effort policy's item- and replicate-specific terminal length L_{ir}^{c}. Let $R_{ir}^{c} = 1$ when both token-exact reasoning prefixes can be reconstructed: high effort either reaches L_{ir}^{c} or has a recoverable terminal boundary before it. For replayable pairs, let $\ell_{ir}^{\mathrm{high},c} = L_{ir}^{c}$ when high effort remains active at the matched horizon and $\ell_{ir}^{\mathrm{high},c} = L_{ir}^{\mathrm{high}}$ when high effort finishes earlier. Under a common medium-effort answer context, define

$$D_{ir}^{c} = Y_{ir}^{c,P_{\mathrm{com}}}(L_{ir}^{c}) - Y_{ir}^{\mathrm{high},P_{\mathrm{com}}}(\ell_{ir}^{\mathrm{high},c}).$$

This definition selects the first L_{ir}^{c} high-effort reasoning IDs when high effort remains active and the complete high-effort reasoning block when it finishes earlier. The complementary all-replayable estimand is

$$\tau_{G,R}^{c} = \mathbb{E}_{i,r}[D_{ir}^{c} \mid R_{ir}^{c} = 1].$$

The high-active subset, in which high effort has not reached a terminal boundary by L_{ir}^{c}, conditions on unfinished high-effort reasoning rather than all replayable pairs. Fixed-checkpoint both-active strata address the narrower comparison between two unfinished policies. These conditional estimands answer different questions from $\tau_{Q,\mathrm{pol}}(B)$.

We retain every matched item–replicate pair with an observed lower-effort terminal outcome. When a high-effort trace cannot be replayed, worst-case bounds set its outcome first to correct and then to incorrect. Greedy final-channel continuations repeat the all-replayable matched-horizon comparison with $\rho = C$ and provide the behavioral check.

A.4 DATASETS AND SAMPLED ITEMS

We use the English-language benchmarks GPQA Diamond (Rein et al., 2024), 198 expert-validated graduate-level multiple-choice questions in biology, chemistry, and physics with four choices (A–D); MMLU-Pro (Wang et al., 2024), 12,032 questions across 14 categories with ten choices (A–J); and Omni-MATH-2 (Gao et al., 2024), 4,181 open-ended problems with free-form answers, introduced by Ballon et al. (2026a). The Qwen numeric/concision experiments use the full GPQA Diamond set and a 500-item MMLU-Pro subset drawn by proportional stratified sampling over the 14 categories with seed 0, so that every budget condition is evaluated on the identical items. The nine-arm Qwen prompt experiment reuses this MMLU-Pro subset.

A.5 MODELS, PROMPTS, AND GENERATION

We evaluate Qwen3-4B, -8B, -14B (Yang et al., 2025) with a 32,768-token context, and gpt-oss-20b, -120b (Agarwal et al., 2025) with a 131,072-token context, all served locally from fixed Hugging Face model versions. Qwen models run in bfloat16 without weight quantization; gpt-oss models use their released MXFP4 quantization.

Full reasoning traces are generated in thinking mode with model-recommended sampling: Qwen3 uses temperature 0.6, top-p 0.95, and top-k 20 with up to 28,000 reasoning tokens; gpt-oss uses temperature 1.0, top-p 1.0, and top-k 50 with up to 126,000 tokens. The three-replicate Qwen budget curve and gpt-oss experiments use three independently seeded global generation replicates. We store full completion token IDs and seeds, require every planned generation to be present, and reject any pair of global replicates whose complete per-item token-sequence collections are identical within a model–benchmark–condition combination. All 23,034 planned generations are present, no two global replicate collections are identical, and the largest incidental item-level exact-match rate within a combination is 0.6%.

The candidate-logit readouts and controls use local Transformers/PyTorch forward passes rather than vLLM's returned log probabilities. Effort-conditioned gpt-oss traces are generated through the chat template with thinking enabled and the reasoning-effort setting low, medium, or high. In the released template this renders `Reasoning: low|medium|high` before the user message and assistant generation prompt. Prompted Qwen3 traces append the tested textual instruction to the system prompt and otherwise leave generation unconstrained, so the external horizon B is applied identically to prompted and unprompted trajectories at probe time. The returned-top-20 readout is retained for the fixed-stop calibration and the wrong-prefix continuation analysis, not for the main estimates (Figure A3).

The numeric/concision prompt is present during treated generation and absent during ordinary generation. Prompt-control generations replace it with the alternatives in Table A9, using the same imposed horizon and the same ordinary-generation baseline within that control cohort. The policy-context readout reconstructs each arm from its stored source-prompt IDs, so the generation-time instruction remains visible when an active prefix is probed. The common-context readout instead uses the ordinary base-prompt IDs for every Qwen arm, replays the arm's exact reasoning IDs, and appends the same neutral closure. Completed reasoning blocks are also replayed under this common context; their original final answers are retained only in the terminal-aware policy-context estimand. Budget-present controls add the numeric/concision prompt only at answer time while holding the source prefix fixed. The token horizon is defined with the source model's tokenizer, so absolute horizons are paired within model and remain descriptive across model families.

A.6 DATA, MODELS, AND SOFTWARE

The GPQA Diamond dataset Rein et al. (2024) is available at `https://huggingface.co/datasets/Idavidrein/gpqa`. The MMLU-Pro dataset Wang et al. (2024) is available at `https://huggingface.co/datasets/TIGER-Lab/MMLU-Pro`. The Omni-MATH-2 dataset is the public release of Ballon et al. (2026a).

The Qwen3 model family Yang et al. (2025) is available at `https://huggingface.co/collections/Qwen/qwen3`. The gpt-oss model family Agarwal et al. (2025) is available at `https://huggingface.co/collections/openai/gpt-oss`.

We used Python 3.13.1 across separate environments. Data processing, analysis, and visualization used *pandas* 3.0.0, *pyarrow* 23.0.0, *scikit-learn* 1.7.2, *SciPy* 1.17.0, *statsmodels* 0.14.6, and *matplotlib* 3.10.8. The original generations used *vLLM* 0.12.0 with *flashinfer-python* 0.5.3. The $R = 64$ answer stages and the replication cohort used *vLLM* 0.21.0 with *flashinfer-python* 0.6.8 and *Transformers* 5.9.0. Local candidate-logit readouts used *PyTorch* 2.11.0 and *Transformers* 5.8.0.

A.7 Trace parsing and forced-answer probes

Qwen3 reasoning is the token sequence between the thinking tags. gpt-oss reasoning is the analysis-channel token sequence, including the channel marker needed to form a valid replay. For the main analyses using stored token IDs, we locate the terminal boundary in the stored completion IDs, exclude all following final-answer IDs, and concatenate the chosen prompt IDs, the first B reasoning IDs or the complete reasoning block, and the closure IDs. No generated reasoning prefix is decoded and retokenized. Where original prompt IDs are available, replay checks compare them with the stored prompt. The gpt-oss generated-answer continuations reconstruct the source-policy prompt and verify its stored token count, as qualified in Section A.1. Boundary placement and source-token hashes are checked separately.

For gpt-oss at the lower-effort matched horizon L_c, a high-effort trace is replayable when it contains at least L_c stored reasoning IDs, whether or not it later reaches a terminal boundary, or when a terminal boundary identifies a completed reasoning block before L_c. A stored sequence that ends before L_c without such a boundary is retained among the matched pairs but has an unavailable high-effort outcome. This rule includes capped high-effort traces whenever their stored reasoning IDs reach the matched horizon. The policy-chosen-horizon analysis uses all replayable pairs; the high-active diagnostic retains only the first case.

The nine-arm Qwen prompt experiment follows the same token-exact construction at $B \in \{512, 2{,}048\}$. All $36{,}000$ generations and $72{,}000$ policy-context replays are present, with no probe errors or context truncations. Cohorts without stored token IDs use text reconstruction: the reasoning text is tokenized with the source model, cut to the requested horizon, decoded, and embedded in the probe prompt. They do not supply token-exact evidence.

The candidate-logit probe contains the single-letter system instruction, question, replayed reasoning prefix, and a family-specific suffix that closes reasoning. For gpt-oss, the suffix closes the analysis channel and opens the final channel. One deterministic forward pass extracts full-vocabulary logits at the first answer position. The valid set contains every supported one-token bare-letter and leading-space spelling; spellings for the same answer are aggregated by log-sum-exp. If p_y is the resulting unnormalized probability for option y, we report valid-option mass $M = \sum_{y \in \mathcal{Y}} p_y$ and the conditional distribution $q_y = p_y / M$. The predicted answer is $\arg\max_y q_y$, conditional log loss is $-\log q_{y^\star}$, and multiclass Brier score is $\sum_y (q_y - \mathbf{1}\{y = y^\star\})^2$. Lower log loss and Brier score are better. Qwen policy-context proper-score contrasts are restricted to paired active prefixes because a retained natural terminal answer has no associated probe distribution. Common-context terminal replay supplies probe distributions for completed reasoning blocks as well.

Each candidate-logit probe records the valid answer-token IDs, missing IDs, coverage, and any probe error. Terminal-prefix agreement is reported in Table A11. In a separate single-run cohort, none of the $2{,}788$ terminal gpt-oss replay prefixes contains a final-channel delimiter (Table A44); this check does not exhaustively cover the three-replicate main cohort. The returned-top-20 calibration is incomplete by construction and is not used for the main estimates (Figure A3). For the six-instruction Qwen prompt panel and the central gpt-oss comparisons, we append the appropriate final-channel closure to the exact stored prefix and greedily generate up to 64 and 16 unconstrained answer tokens, respectively. The separate literal Qwen cap-then-answer validation uses a 16-token answer allowance. Missing or malformed answer letters count as incorrect and as parse failures. These continuations decode answers deterministically from independently sampled reasoning prefixes, so the reasoning replicate is the stochastic unit. Terminal-prefix readouts reproduce the naturally sampled final answer for 98.3–100.0% of completed prefixes in the terminal-agreement check.

The Omni-MATH-2 analyses use free-form answers judged with `gpt-5-mini-2025-08-07`, following Ballon et al. (2026a). The Qwen comparison uses a 200-item subset and includes item–horizon comparisons for which both prompt and surprise answers were judged. The gpt-oss comparison draws 500 items using Omni-MATH's native difficulty metadata (Gao et al., 2024); the analyzed

model-specific paired cohorts contain 475 items for gpt-oss-20b and 490 for gpt-oss-120b. All gpt-oss prefixes are replayed under a common medium-effort final-answer context, and the analysis includes all $15{,}440$ judged answers across the matched-horizon and fixed-checkpoint comparisons. These comparisons use one rollout per condition and therefore do not characterize variation across sampled generations. An independent full-trace judging pass agrees on 94–99% of items across the five models. Structural-marker annotations use the same snapshot with the released JSON-schema prompt.

A.8 Readout contexts and answer availability

We run policy- and common-context readouts to separate the full interface contrast from differences carried by the generated reasoning prefix. The Qwen policy-context readout preserves the stored condition-specific prompt. The Qwen common-context readout uses the ordinary base prompt and neutral closure for every arm and replays completed as well as active reasoning blocks. A paired answer-context control then reintroduces the numeric/concision prompt only at answer time while holding the bundled-prompt prefix fixed. For gpt-oss, the policy-chosen-horizon readout uses a common medium-effort answer context for every low-, medium-, and high-source prefix. Additional controls cross the source effort with the effort setting used only at readout. We treat candidate-logit scoring as an option-normalized, deterministic paired instrument for answer availability at a standardized prefix. Greedy final-answer continuations provide the corresponding behavioral readout.

A source-context readout reconstructs the source prompt and reasoning prefix at the answer point instead of placing the prefix in a neutral context. Qwen surprise-stop traces keep the budget absent, Qwen bundled-prompt traces retain the numeric-budget instruction, and gpt-oss traces retain the source effort setting. This text-based recomputation approximates the trajectory's original prompt context rather than recovering the generation-time KV cache.

To test when answers become available in the trajectory, we run a dense readout over the generated prefixes at checkpoints $0, 64, \ldots, 512$. For gpt-oss, the source prefixes come from low, medium, and high effort traces, and all are read with a common medium-effort readout. For Qwen3-14B, the source prefixes compare bundled-prompt and surprise-stop traces at the $B = 512$ condition, and the readout uses a neutral budget-free closure. At each checkpoint we extract candidate-letter probabilities from full-vocabulary next-token logits and compute the gold-answer log-odds, $\log p(y^\star) - \log \sum_{y \neq y^\star} p(y)$. Natural termination is treated as an absorbing state. For a trace that terminates at length $L_{ic} < 512$, we carry its terminal forced-answer distribution forward, so that $q_{ic}(t) = q_{ic}(L_{ic})$ for every $t \geq L_{ic}$. The resulting terminal-aware answer-availability AUC is a token-normalized trapezoidal average that includes both evidence accumulated before termination and the practical consequence of having completed the answer early. We also report retrospective stable correctness, which counts a checkpoint as correct only if the predicted answer remains correct at all later checkpoints through 512.

As a sensitivity analysis, we recompute the terminal-aware answer-availability AUC over 0–128 tokens after retaining only items for which both compared traces are still reasoning after 128 tokens ($L > 128$). We call this an active-pair sensitivity analysis rather than a separately identified causal estimate, because conditioning on survival changes the target comparison.

A.9 Literal Qwen cap-then-answer validation

The literal Qwen cap-then-answer validation follows the documented two-stage thinking-budget procedure (Qwen Team, 2025) while retaining the paired item–replicate design of the candidate-logit analysis. The design crosses GPQA Diamond (198 items) and the main MMLU-Pro subset (500 items) with $B \in \{512, 2{,}048\}$, hidden/surprise and numeric/concision-visible arms, and the same three generation replicates. In the first call, vLLM generates the reasoning block with the Qwen sampling settings and an external maximum of B tokens, stopping early if `</think>` appears. If generation hits the cap, we append the Qwen limited-time closure and close the thinking block; otherwise we retain the naturally closed state. In the second call, the model greedily generates up to 16 answer tokens. Malformed or missing choice labels are parse failures and count as incorrect. We store the full reasoning- and answer-stage token IDs, hash the reasoning and answer stages together, and reject a benchmark–arm–budget combination if two complete replicate collections are record-identical across its items. Accuracy contrasts pair arms within item, replicate, and budget. The reported intervals average the three observed replicate outcomes within item and bootstrap items;

replicate-specific, two-way item × replicate, and crossed-intercept calculations are sensitivities. Candidate-logit deltas are included only as a readout comparison.

A.10 SCRIPTED POSITIVE CONTROL

To check that the readout at a shared horizon detects a known early-answer trajectory, we construct a scripted synthetic positive control with 96 balanced four-choice items. The question contains no answer information. All options are generic labels A–D, and the gold label is assigned by a deterministic balanced key. The long policy uses a neutral reasoning prefix in which the candidate-answer line appears after the first 64 source tokens but before 512 source tokens. The short policy uses a prefix beginning with `candidate answer: X`, where `X` is the gold label. A short $B = 512$ policy reuses the long trace, so the positive control should be large at 64 tokens and vanish once both traces reveal the candidate answer.

We generate probes directly from these scripted reasoning prefixes, truncate them with the Qwen tokenizer, and apply the same candidate-logit answer-token readout used for the paired multiple-choice contrasts and controls. Accuracy moves from 25.0% for the long policy to 100.0% for the short policy at $B = 64$; both reach 100.0% at $B = 512$ (Table A43). This positive control validates the truncation and readout on a known text-level policy difference. It is not evidence that Qwen3-14B or gpt-oss spontaneously condition on exact numeric budgets.

A.11 TRAJECTORY DIAGNOSTICS AND CONTROLS

To test whether a condition can be distinguished from the early reasoning, we train a logistic-regression classifier to predict the condition from the first k reasoning tokens of a trace, for $k \in \{64, 128, 256, 512, 1024, 2048\}$. The input is the surface text of the first k generated reasoning tokens and nothing else: no question, no answer, no trace length, and no model identity. Traces shorter than k are excluded at that k rather than padded, so termination or residual length is not provided as an implicit feature. We use two feature sets, TF-IDF over word n-grams (1–3) and character n-grams (3–5), and a set of length-normalized style counts (backtracking and verification cue words, punctuation, operator and digit density, answer-letter mentions, mean word length). The label is the effort setting (low, medium, high) for gpt-oss and the bundled-prompt condition (bundled prompt versus surprise-stop) for Qwen3. We evaluate with 5-fold cross-validation grouped by item, so the same question never appears in both train and test, and compare against a label-shuffle baseline of 20 permutations.

We analyze the generated reasoning text rather than treating prompt-boundary hidden-state decodability as evidence that the reasoning policy changed. The effort or budget instruction is present in the prompt, whereas the surface text is generated by the policy itself. We therefore use text classifiers for the early-policy diagnostic.

To characterize how the early reasoning differs across conditions, we compute length-normalized structural markers on the first $\min(256, L)$ reasoning tokens of each trace: the position of the first computational cue, the position of the first answer assertion, the rates of backtracking, verification, and alternative-strategy cue phrases per thousand words, characters per word, and digit and operator density. Raw trace length is not among the reported markers. Markers are compared between paired conditions with bootstrap intervals over items on the same GPQA Diamond and stratified MMLU-Pro analysis subsets used for the main probe contrasts. These are deterministic lexical patterns and capture surface form rather than meaning. To validate them, we annotate a stratified sample of 250 trace excerpts (50 per condition) with GPT-5 mini for the same constructs and correlate the cue-based markers with the annotations (n=204 matched). The backtracking and alternative-strategy markers agree with the annotations (Spearman ρ=0.32 and 0.19, respectively; both $p < 0.01$), whereas the verification marker is weaker and not significant (point-biserial r=0.11, $p = 0.109$), and the two position-based markers, restatement length and first-answer commitment, do not validate. We base our structural interpretation on the digit and operator density and the alternative-strategy marker, which are either objective character statistics or annotation-supported.

To test whether the recovered accuracy is instance-specific, we re-probe each surprise-stop prefix against three length-matched controls at the same budget B: a *random* prefix of B tokens sampled uniformly from the vocabulary excluding special tokens; a *swap* prefix taken as the first B reasoning

tokens of a different question's trace; and a *shuffle* prefix that permutes the first B original reasoning tokens. Swap donors must be distinct and contain at least B source tokens. We exclude 31 swap prefixes whose donor traces are shorter than B and restrict all pooled contrasts to the remaining common four-arm item–budget cohort. The controls isolate, respectively, context length and position, coherent but irrelevant reasoning, and token identity without order.

A.12 STATISTICAL INFERENCE

The three-replicate candidate-logit comparisons use three independently sampled global generation replicates. Conditions are paired within item and replicate. The reported intervals condition on the three trajectory sets observed in this experiment: we first average the three paired replicate outcomes within item and then draw $10{,}000$ bootstrap resamples of items. Bootstrap resampling uses NumPy seed 0 by default, and every reported interval uses a fixed, recorded resampling seed. We report each replicate separately with an item bootstrap. A paired two-way item $\times$ replicate bootstrap and a crossed item–replicate variance decomposition are sensitivity analyses. With only three global replicate levels, those sensitivities do not provide precise population-over-seed inference.

For the model-based sensitivity analysis, we fit the paired differences with crossed item and replicate random intercepts. Variance components are estimated by method of moments from complete matrices of items by replicates; uncertainty for the grand mean uses the item, replicate, and residual components with a Satterthwaite t approximation. With three global replicates, this calculation is a diagnostic of sensitivity to trajectory-set variation rather than a precise variance-component analysis.

The nine-arm Qwen prompt experiment uses eight item-specific replicates for each of 500 MMLU-Pro items. A deterministic hash of the item ID, replicate number, and base seed defines one generation seed for each item–replicate pair; the same seed is reused across all nine prompt arms, and all $4{,}000$ item–replicate seeds are distinct. The arms are hidden horizon; numeric/concision prompts announcing 512 or $2{,}048$ tokens; advance exact-stop notices announcing the same two horizons; generic limited-budget wording; number-only wording; a concise/early-answer instruction; and an irrelevant-number placebo. The reported intervals use a nested paired bootstrap that first resamples items and then resamples the eight paired replicates within each selected item. The arm–horizon family is covered by simultaneous intervals obtained from the maximum absolute deviation across the treatment-versus-ordinary contrasts in each bootstrap draw. Direct prompt-to-prompt contrasts use the same paired draws. The direct diagonal statistic averages the advantage of announcing 512 rather than $2{,}048$ tokens when the imposed stop is 512 with the reverse contrast when the imposed stop is $2{,}048$. A positive value would indicate that matching the announced and imposed horizons helps.

For the three-replicate Qwen candidate-logit curve, each item's paired outcomes are first averaged over the observed replicates. The main bootstrap resamples items and preserves each selected item's complete budget curve. The simultaneous 95% band uses the 95th percentile of the maximum absolute deviation from the estimated curve. A scalar summary integrates the accuracy difference over $\log_2 B$ with trapezoidal weighting and divides by the observed log-budget span; its percentile interval is computed from the same item-bootstrap draws. A two-way item $\times$ replicate version of both summaries is retained as a sensitivity analysis. The ± 3 percentage-point margin is a descriptive scale and is not used as an equivalence threshold.

For the gpt-oss matched-horizon comparison, we retain every matched item–replicate pair with an observed lower-effort terminal outcome and bound the effect of unreplayable high-effort traces. Let $\mathcal{O}_c$ and $\mathcal{M}_c$ denote the replayable and missing-high sets, let $N_c = |\mathcal{O}_c| + |\mathcal{M}_c|$, and write the observed completed lower-effort outcome as $Y_{ir}^{c,N}$. Worst-case binary bounds set every missing

high-effort outcome first to correct and then to incorrect:

$$\underline{\tau}_G^c = \frac{1}{N_c}\left[\sum_{(i,r)\in\mathcal{O}_c} D_{ir}^c + \sum_{(i,r)\in\mathcal{M}_c} (Y_{ir}^{c,N} - 1)\right],$$

$$\overline{\tau}_G^c = \frac{1}{N_c}\left[\sum_{(i,r)\in\mathcal{O}_c} D_{ir}^c + \sum_{(i,r)\in\mathcal{M}_c} Y_{ir}^{c,N}\right].$$

Several conditions in the older trace archives have identical decoded responses across their stored runs, so those runs are not independent; the archives serve only the descriptive length, marker, classifier and text-reconstructed prompt analyses. Prompt-control, readout-context, crossed-horizon, and subset-replication cohorts analysed from decoded text are excluded from the token-exact main estimates. The Omni-MATH-2 comparisons use one rollout per condition.

Single-run binary matched-pair checks use item-bootstrap intervals. Where McNemar's test is reported, it is exact when the discordant count is below 25 and otherwise uses the continuity-corrected χ^2 approximation (McNemar, 1947). Repeated-budget prefix-control summaries first average paired outcomes over budgets within item and then use an item-cluster bootstrap. Every budget curve reports the total item count, the number surviving to B, the survival rate, and both survivor-only and terminal-aware policy accuracy.

A.13 Replication, interface comparisons, and diagnostic analyses

A.13.1 Text-reconstructed Qwen subset replication and prompt comparisons

The text-reconstructed MMLU-Pro subset replication compares the 500-item subset used in the main design with a second category-stratified 500-item subset that excludes those items. Both subsets compare the numeric/concision and surprise-stop conditions at $B = 512$ over three stored runs, and the independent subset also compares the numeric/concision and concise/early-answer conditions with the same surprise-stop cohort at $B = 2{,}048$. Accuracy and survival use the candidate-logit readout. Intervals resample items and stored runs in a paired two-way bootstrap. These cohorts were analysed from decoded text rather than stored token IDs, and the independence of their runs was not confirmed, so the estimates describe the observed runs and do not support generalization across generation seeds.

The text-reconstructed prompt-control comparison evaluates eight Qwen3-14B instructions against the same surprise-stop cohort at $B = 512$ on GPQA Diamond and MMLU-Pro. Each estimate averages three stored runs within item before an item-cluster bootstrap. Table A34 reports terminal-aware accuracy and median full-trace length; paired log-length contrasts are included in the companion data release. These cohorts were also analysed from decoded text rather than stored token IDs.

A.13.2 Numeric wording with gpt-oss effort

The within-model gpt-oss comparison crosses the numeric-budget instruction with low, medium, and high effort; $B \in \{512, 2{,}048, 8{,}192, 16{,}384\}$; both gpt-oss model sizes; and GPQA Diamond or MMLU-Pro. The directly matched comparison adds the numeric wording to each effort setting and evaluates both conditions at $B = 512$ (Table A42). Longer announced-budget comparisons lack a directly matched effort-only baseline because the candidate-logit effort-only comparison for this interaction covers only $B = 512$, and low-effort traces often terminate before later checkpoints.

A.13.3 WRONG-PREFIX CONTINUATION

The wrong-prefix continuation analysis selects natural Qwen3-14B traces with an incorrect parsed final answer, retains the first B source-tokenized reasoning tokens, and presents that prefix with the same question. The answer-now condition closes reasoning immediately with the forced-answer suffix. Continuation conditions instead allow R additional reasoning tokens before the forced answer, with the remaining budget either hidden or stated in the prompt. Both conditions use the returned-top-20 readout, whose incomplete option coverage applies symmetrically. We evaluate $B \in \{512, 2{,}048, 8{,}192\}$ and $R \in \{512, 2{,}048, 8{,}192, 32{,}768\}$ on GPQA Diamond and the stratified MMLU-Pro subset. The displayed $R = 32{,}768$ conditions are limited by the remaining context window (Table A35).

A.13.4 ACTIVATION READOUT OF THE ANNOUNCED BUDGET

The activation analysis tests whether Qwen3-14B hidden states retain the announced numeric budget. For each item with one natural trace and all four numeric-budget traces, we feed the prompt and the first $k \in \{0, 64, 256, 512, 1024, 2048\}$ generated reasoning tokens. We random-project hidden states to 256 dimensions at layers 0, 10, 20, 30, and 39, then fit logistic-regression readouts with item-grouped cross-validation. One readout predicts whether the trace uses the numeric/concision prompt; the other predicts which of the four announced budgets generated a treated trace. Each is compared with 20 label-shuffle baselines. A focused check uses 100 shuffles and $C \in \{0.1, 1, 10\}$ for the post-start exact-budget comparisons (Table A36).

A.14 QWEN REPLICATION COHORT

A replication tests the six-instruction generated-answer panel on questions that played no role in prompt selection or the answer-allowance check. Its procedure was fixed before inference and is included with the analysis code. A category-stratified 500-item MMLU-Pro subset was drawn with seed 20260923 from the questions outside both the main subset and the text-reconstructed independent subset. Each item received eight independently seeded generations per instruction with master seed 20260923, using the same system prompt composition, sampling settings and 28,000-token reasoning limit as the prompt-selection cohort, giving 24,000 traces. Exact prefixes at 512 and 2,048 tokens were replayed in the source-policy and common contexts with the neutral closure, and every prefix, including completed reasoning blocks, received a fresh greedy $R = 64$ answer, giving 96,000 outcomes scored with the same parser. Parsing failures count as incorrect. The contrasts, matching definition and 10,000-draw nested item–replicate bootstrap with seed 0 are those of the prompt-selection cohort. Generation ran as eight single-GPU tasks of 34 to 56 minutes and the answer stage as one 41-minute task on H200 GPUs. Among pairs whose two answers both parse, the concise advantage is $+3.0$ points ($[+1.3, +4.7]$) at 512 tokens and $+0.5$ ($[-0.8, +1.7]$) at 2,048, as in the prompt-selection cohort. When completed reasoning blocks keep their natural answers instead of receiving fresh ones, the concise gains are $+3.5$ ($[+1.8, +5.2]$) at 512 tokens and $+1.8$ ($[+0.4, +3.2]$) at 2,048, and the matching contrasts are unchanged. Among pairs that both policies leave unfinished and whose two answers both parse, the concise advantage is $+3.5$ ($[+1.2, +5.9]$) at 512 tokens in the source-policy context and $+4.2$ ($[+2.0, +6.4]$) in the common context, but $+3.2$ ($[-0.7, +7.3]$) and $+1.9$ ($[-2.0, +6.0]$) at 2,048, where ordinary generation fails to parse in 16% of unfinished pairs against 5% for the concise instruction (Table A39c). At natural completion, the concise instruction reaches 75.6% against 76.6% for ordinary generation (-1.0 points, $[-2.3, +0.2]$) while mean generated tokens fall from 2,358 to 1,626 (-732, $[-833, -639]$; Table A39d). We keep the common-seed pairing across instructions, and the jointly unfinished contrasts are conditional on that pairing. Re-pairing the eight concise replicates against the eight ordinary replicates within each item 10,000 times keeps the 512-token jointly unfinished contrast between $+3.5$ and $+4.6$ points in the source-policy context and between $+4.1$ and $+5.3$ in the common context, positive in every draw (Table A40). Of the replication's 2,602 parse failures, 2,600 exhaust the 64-token answer allowance, so they are continued reasoning in the answer channel rather than malformed labels.

We also generated four further instructions on the same items with the same paired seeds, adding the announced number to the concise instruction either as the advance notice of an exact stop or as the numeric/concision prompt (Table A41). The advance notice leaves the concise gain intact: relative to the concise instruction alone, generated-answer accuracy changes by $+1.1$ points ($[-0.3, +2.5]$) at 512 tokens and -0.1 ($[-1.3, +1.1]$) at 2,048, completion by the announced horizon rises by 4.7

and 2.2 points, and the combined instruction is $+2.4$ points ($[+0.6, +4.1]$) more accurate at 512 tokens than the notice alone. Adding the numeric/concision prompt instead lowers accuracy by 3.0 ($[-4.9, -1.0]$) and 4.4 ($[-6.1, -2.7]$) points, although 48.6% of its reasoning blocks finish by 512 tokens against 34.0% for the concise instruction. Matching the announced number to the evaluation horizon again adds little within either family, $+0.1$ ($[-0.6, +0.8]$) and $+0.6$ ($[-0.1, +1.4]$) points, and the common-context replay gives similar estimates. Generation ran as eight single-GPU tasks of 26 to 42 minutes and the answer stage as one 26-minute task.

B Additional results and diagnostic analyses

This section reports scope, replication, and diagnostic analyses. These analyses test the boundaries of the main findings; they do not replace the token-exact paired estimates.

B.1 Readout validity controls

Stopped prefixes reveal increasingly accurate, question-specific answers as the imposed horizon grows (Figure A1a,b,d). For Qwen3-14B on MMLU-Pro, terminal-aware policy accuracy rises from 50.4% without reasoning to 60.0%, 70.8%, and 76.6% at $B = 512$, 2,048, and 8,192, approaching the cohort-matched full-trace accuracy of 76.8% ($384/500$; Table A10). Across all five models, saturation follows natural reasoning length: the shorter gpt-oss traces plateau by $B = 2{,}048$, while Qwen remains on the rising part of the curve and smaller models show a few late-horizon reversals or near-ties (Figure A4). The calibration curves and natural references share one natural-trace cohort. The Qwen prompt comparison below uses separately generated, replicate-matched cohorts (Section A).

Survival falls as terminal-aware accuracy rises, making survivor-only accuracy a selected comparison (Figure A1c and Table A10). The model's decision to continue reasoning changes which items remain at a large horizon, so survivor-only accuracy is flat or non-monotone even as the terminal-aware curve improves. At $B = 8{,}192$, only 18.7% of Qwen3-14B and 9.1% of gpt-oss-20b GPQA trajectories are still reasoning. We therefore report survival with every horizon curve rather than conflating the evaluation point with the selected set of items that reaches it.

The candidate-logit probe also reproduces completed answers without replaying final-channel text. A separate validation cohort yields 98.3–100.0% agreement between replayed-prefix and parsed natural answers across its model–benchmark–policy combinations (Table A11). A single-run check also finds no final-channel delimiter in 2,788 terminal gpt-oss replay prefixes (Table A44). These cohorts validate terminal replay but do not exhaustively check the three-replicate token-exact analysis. The fixed-stop and prefix-control probes have no errors or missing valid answer-token IDs. Answers asserted inside the reasoning remain part of the measured behavior.

Length-matched controls confirm that the recovered answer depends on the question-specific prefix (Figure A1d). Swapping in reasoning from another question reduces accuracy by 18.1 points for gpt-oss-20b and 17.8 points for Qwen3-14B, while shuffled-token and random-token controls fall between the swap and original prefixes (Figure A5). These checks support comparisons between generated prefixes at matched evaluation points.

B.2 Qwen scope, replication, and diagnostics

The full candidate-logit completion-stratum and probability-score summaries accompany the prompt comparisons in Tables A21 and A27.

The literal cap-then-answer validation of the numeric/concision prompt reproduces the benchmark split but no broad gain (Figure A13e and Table A13). A text-reconstructed four-budget crossed analysis shows no diagonal matching-number pattern, and a separate four-horizon concise/early-answer comparison likewise shows no broad advantage (Figure A6 and Table A32). In a pooled MMLU-Pro $B = 512$ decomposition, category differences range from -7.1 to $+7.8$ points (Table A32). A single-rollout Qwen3-8B comparison also shows no broad same-horizon gain (Table A31). A 200-item open-ended Omni-MATH-2 comparison finds differences of $+3.5$, $+2.0$, and -4.1 points across the three tested horizons, with every interval spanning zero (Table A12). A scripted control moves

from 25.0% to 100.0% at $B = 64$ and reaches 100.0% in both arms by $B = 512$, confirming that the readout detects a known early-answer policy (Table A43).

A text-reconstructed MMLU-Pro subset replication broadens the item sample, but it was analysed from decoded text rather than stored token IDs and therefore does not enter the token-exact main analysis. On a second 500-item subset, the numeric/concision contrast is $+2.5$ points at $B = 512$ ($[+0.4, +4.8]$), and pooling the original and independent subsets gives $+2.1$ points ($[+0.6, +3.6]$). On that independent subset at $B = 2{,}048$, the numeric/concision estimate is -0.2 points ($[-2.6, +2.2]$), whereas the concise/early-answer estimate is $+2.7$ points ($[+0.5, +5.0]$; Table A33). A separate text-reconstructed, eight-instruction prompt-control comparison at $B = 512$ likewise shows small, mixed GPQA estimates and larger early MMLU-Pro point estimates, with the concise/early-answer instruction again largest. Its clearest pattern is instruction-dependent reasoning length rather than a consistent accuracy gain (Table A34).

Two diagnostic analyses ask how much the early part of a trace constrains what follows. Continuing a Qwen prefix from a trace with an incorrect natural final answer raises rescue from 20.0–30.1% under immediate answering to 26.5–38.2% with a long hidden continuation, but 35.8–55.8% of continuations remain anchored to the original wrong answer and stating the remaining budget gives no consistent benefit (Table A35). Hidden-state readouts recover the exact announced budget at the prompt boundary with balanced accuracy 0.938 on GPQA Diamond and 0.905 on MMLU-Pro, then mostly fall to label-shuffle levels after reasoning begins. Binary bundled-prompt membership remains modestly decodable at most post-start token positions, with balanced accuracy from 0.588 to 0.700 (Table A36). These analyses characterize answer rescue, anchoring, and persistence of prompt information; they are not estimates of the matched-horizon effects.

B.3 GPT-OSS SCOPE AND DIAGNOSTIC ANALYSES

The terminal-aware candidate-logit comparison at $B = 512$ is shown in Figure A9. Matched-horizon arm accuracies and replay coverage are reported in Table A29.

Replicate-specific, two-way, and crossed-intercept calculations agree with the policy-selected-horizon result (Table A19). Answer-context, source-context, and closure checks at the reported checkpoints do not reverse the family-level pattern (Tables A20 and A28). In model-specific paired cohorts of 475 items for gpt-oss-20b and 490 for gpt-oss-120b from a difficulty-stratified 500-item Omni-MATH-2 draw, completed lower-effort answer accuracy exceeds matched-horizon high-effort answer accuracy by $+33.5$ to $+42.0$ points across four comparisons; fixed-checkpoint comparisons are reported in Table A31, but the single rollout does not establish that the estimate generalizes across sampled generations (Table A12). Dense probes place the lower-effort answer earlier in the trajectory: terminal-aware AUC differences are positive in all eight gpt-oss comparisons, whereas jointly active early-prefix differences are much smaller (Table A26 and Figure A7). Surface diagnostics show more digits and operators, fewer explicit alternative-strategy cues, and effort classification accuracy of 0.70 from the first 64 tokens (Figures A10 and A11). These item-conditional diagnostics are not used to compare effect magnitudes across the two case-study estimands.

Adding numeric-budget wording to the gpt-oss effort instruction produces no broad $B = 512$ advantage over effort alone (Table A42). Ten of the twelve intervals span zero. The gpt-oss-20b medium-effort difference on GPQA Diamond is -5.6 points ($[-10.6, -0.5]$); the gpt-oss-20b high-effort difference on MMLU-Pro is $+2.7$ points, with a rounded lower endpoint of zero ($[+0.0, +5.5]$). The candidate-logit effort-only baseline for this interaction ends at $B = 512$, leaving its longer announced-budget contrasts unmatched.

B.4 ANSWER-TOKEN PROBABILITY QUALITY

A better selected answer need not give a better answer distribution under the readout (Figure A12f). Brier estimates favor lower effort in seven of eight terminal-horizon comparisons, with intervals excluding zero in five. Conditional log loss favors high effort in all eight, with intervals excluding zero throughout (Table A23). Incorrect completed reasoning can concentrate answer-token mass on the stated answer, producing near-zero correct-option probabilities and large log loss. This may partly reflect faithful reproduction of an asserted answer. The result concerns the diagnostic distribution, not calibrated confidence in task correctness; probability-tail details remain in Table A30.

C Appendix Figures

The appendix figures are organized by the question each analysis addresses. Figures A3 to A5 characterize readout sensitivity and natural trace lengths and test whether recovered information depends on the question-specific prefix. Figures A6 to A7 test horizon matching and show when answer evidence becomes available. Figures A8 to A9 compare completed-trace and fixed-checkpoint gpt-oss accuracy, while Figures A10 to A11 characterize how effort and prompting alter early traces. The crossed-horizon Qwen cohorts in Figure A6 were analysed from decoded text rather than stored token IDs. They provide a horizon-pattern check and do not enter the token-exact effect estimates.

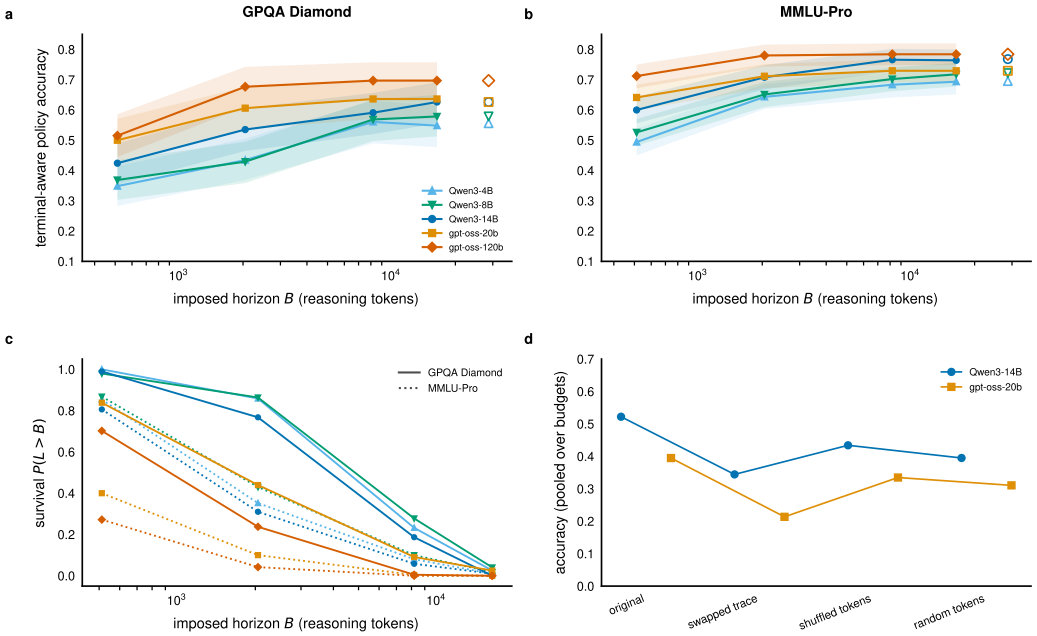


Figure A1: **Longer stopped traces reveal more question-specific answer information.** Surprise-stop probes use candidate-logit readouts on GPQA Diamond (n=198) and MMLU-Pro (n=500); colors and markers denote models. **a,b**, Terminal-aware accuracy approaches the cohort-matched natural full-trace reference as the imposed horizon B grows; bands are 95% item-bootstrap intervals. **c**, Survival $P(L > B)$ falls with B, making late survivor accuracy a selected comparison. **d**, Original Qwen3-14B and gpt-oss-20b prefixes outperform length-matched swapped-trace, shuffled-token, and random-token controls. Figure A3 gives the returned-top-20 sensitivity analysis.

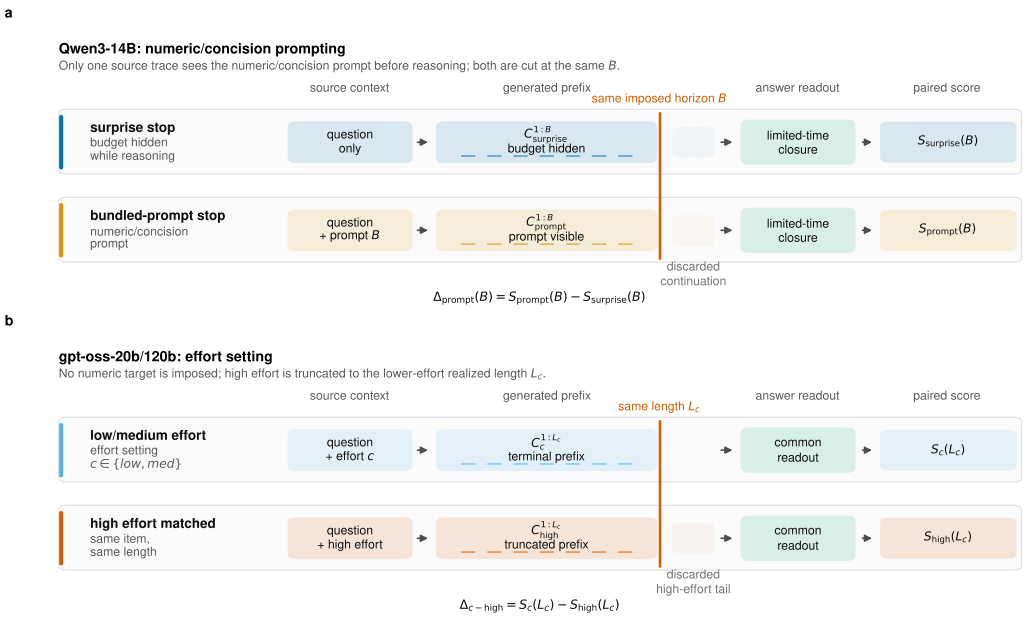


Figure A2: **The candidate-logit comparisons use two matched-horizon designs. a**, Qwen3-14B ordinary and numeric/concision-prompt traces face the same imposed horizon B. Completed traces keep their natural answers; unfinished traces are probed in their policy context, with common-context and both-active diagnostics holding the answer-time prompt fixed. **b**, A completed low- or medium-effort gpt-oss block is compared with high effort at the same itemwise lower-effort length L_c under a common medium-effort answer context. Completed high-effort blocks are replayed when they finish before L_c. Replayed prefixes exclude original final-answer tokens (Table A8).

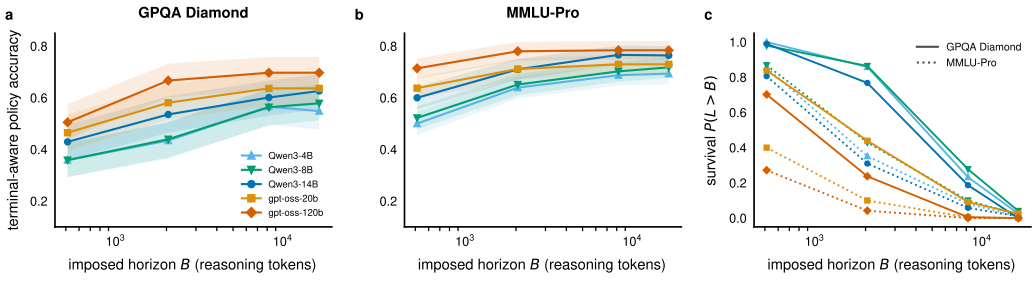


Figure A3: **A returned-top-20 readout reproduces the qualitative fixed-stop pattern.** Terminal-aware policy accuracy (**a**, GPQA Diamond; **b**, MMLU-Pro) and survival (**c**) for surprise-stop probes under a returned-top-20 readout, which takes the argmax over the option letters present in the returned top-20 log-probabilities and has incomplete option-letter coverage (Section A). The full-vocabulary candidate-logit readout (Figure A1) is used for the candidate-logit analyses.

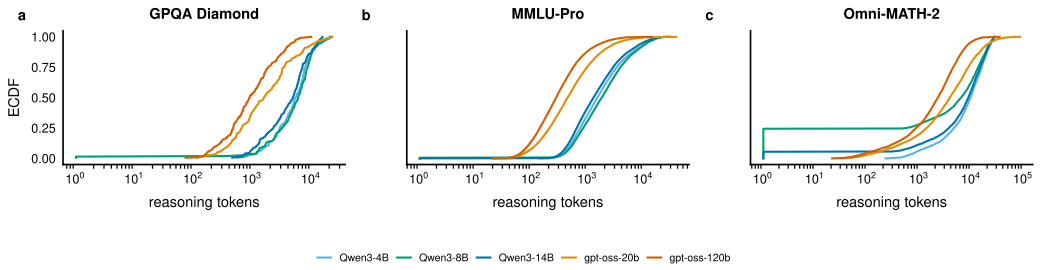


Figure A4: **Natural reasoning-length distributions by model and benchmark.** Empirical cumulative distribution of reasoning-trace token length for the five models on GPQA Diamond, MMLU-Pro, and Omni-MATH-2. Natural length varies severalfold across models and benchmarks, which sets where the imposed-horizon curves of Figure A1 saturate. Panel **c** includes $980/4{,}011$ Qwen3-8B and $223/4{,}039$ Qwen3-14B responses that immediately close an empty `<think></think>` block. We count each empty block as one source-model token and exclude the following final-answer text from reasoning length.

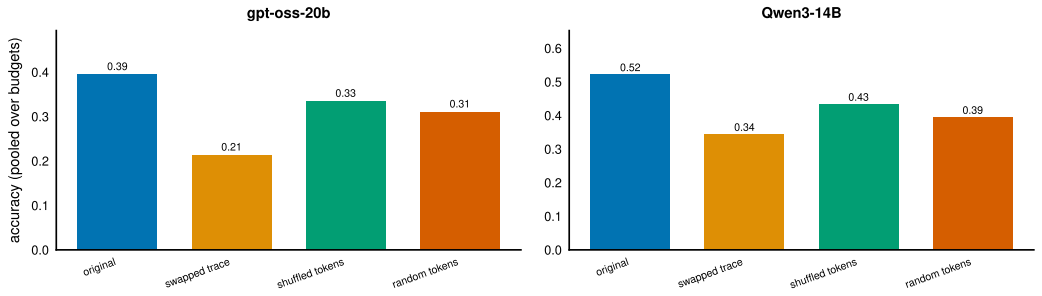


Figure A5: **Truncation-recovered accuracy is instance-specific: the original prefix beats length-matched controls.** Accuracy of the surprise-stop probe under the original budget-B prefix and three length-matched controls, pooled over budgets on the common four-arm item–budget cohort, for gpt-oss-20b and Qwen3-14B on GPQA Diamond and MMLU-Pro. The swapped-trace control (coherent reasoning from a different question) is the weakest, below the shuffled-token and random-token controls, and all sit below the original. Inference averages paired original-minus-control outcomes over budgets within item and uses an item-cluster bootstrap plus item-level sign-flip randomization test.

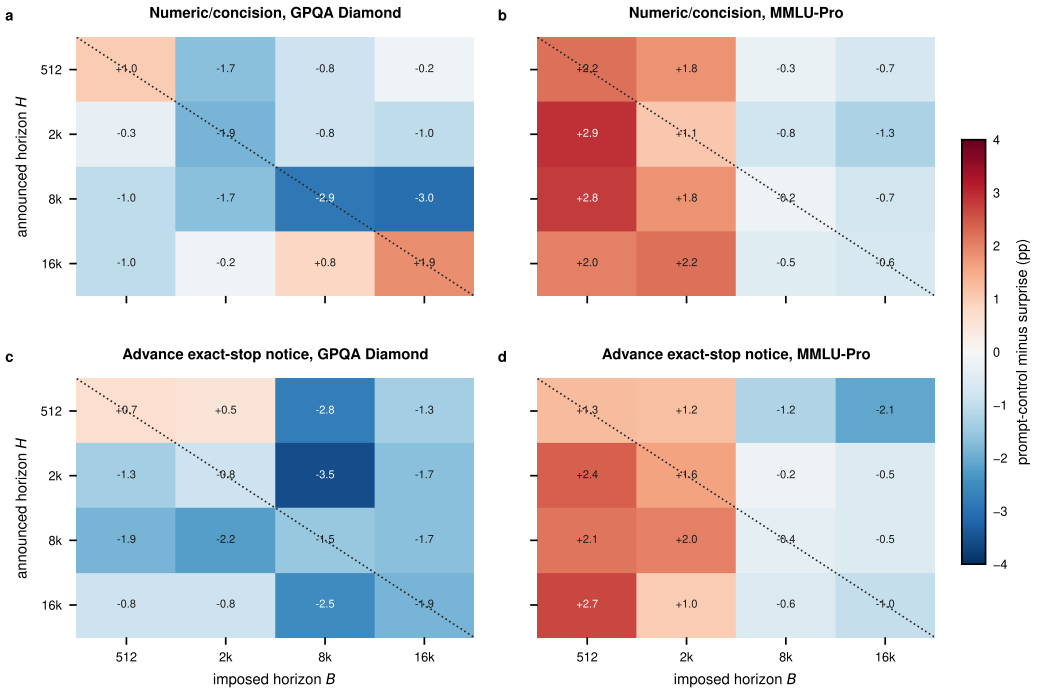


Figure A6: **Crossing announced and imposed horizons shows no diagonal matching-number pattern.** Heatmaps show prompt-control minus surprise-stop accuracy deltas in percentage points for Qwen3-14B under the candidate-logit readout. Rows vary the announced horizon H in the prompt, columns vary the imposed horizon B, and dotted diagonals mark $H = B$. **a,b**, Numeric/concision prompt for GPQA Diamond and MMLU-Pro. **c,d**, Advance exact-stop notice for the same benchmarks. MMLU-Pro positives concentrate at early imposed horizons, while GPQA remains small or negative across the grid. These four-budget and GPQA cohorts were analysed from decoded text rather than stored token IDs; the paired MMLU-Pro experiment likewise finds no clear matching-number benefit at two imposed stops (Table A14).

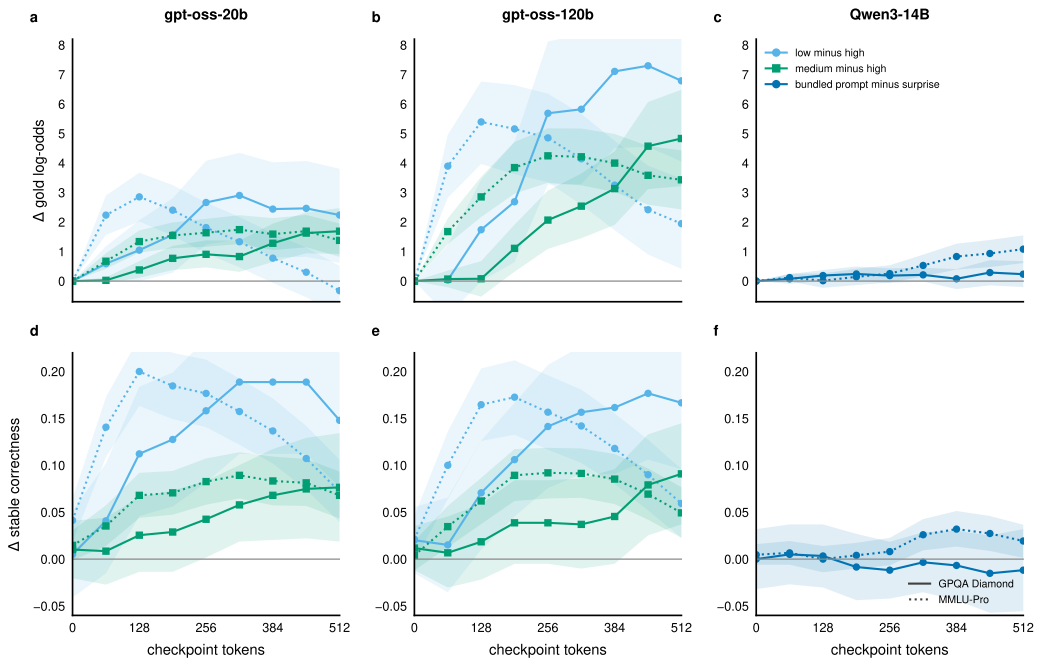


Figure A7: **Shorter gpt-oss effort settings make answers available earlier under the candidate-logit readout.** Curves show paired condition differences at dense checkpoints through $B = 512$; solid lines are GPQA Diamond, dotted lines are MMLU-Pro, and shaded bands are pointwise 95% paired-bootstrap intervals over items. **a–c**, Difference in option-normalized gold-answer log-odds for gpt-oss-20b, gpt-oss-120b, and Qwen3-14B, computed from full-vocabulary next-token logits over every valid answer-token ID. **d–f**, Difference in retrospective stable correctness, where a checkpoint is counted correct only if all later checkpoints through 512 remain correct. The gpt-oss contrasts compare low or medium source prefixes against high source prefixes under a medium-effort readout; Qwen compares bundled-prompt and surprise prefixes under a neutral budget-free closure.

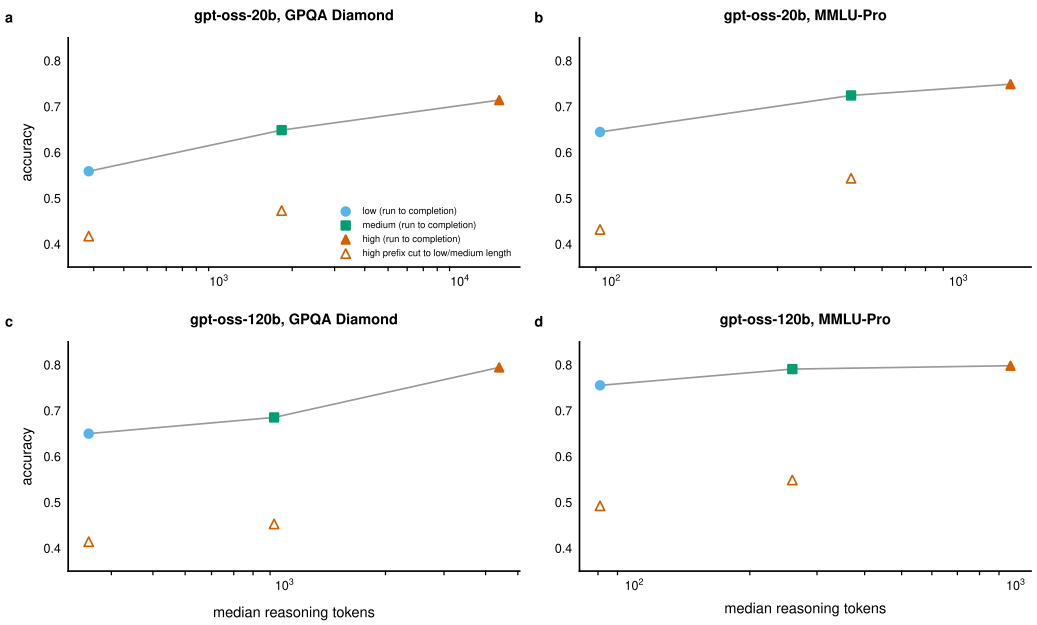


Figure A8: **Completed gpt-oss effort settings trade accuracy against median trace length.** Results for gpt-oss-20b (**a,b**) and gpt-oss-120b (**c,d**) on GPQA Diamond and the stratified MMLU-Pro subset pool three generation replicates. When run to completion, the three effort settings define the observed relationship, while high-effort readouts at the itemwise low- or medium-effort terminal horizon sit below it; a completed high-effort reasoning block is replayed when it finishes before that horizon. Because the horizontal axis uses medians, this figure describes typical trace length rather than arithmetic-mean token use or expected API cost.

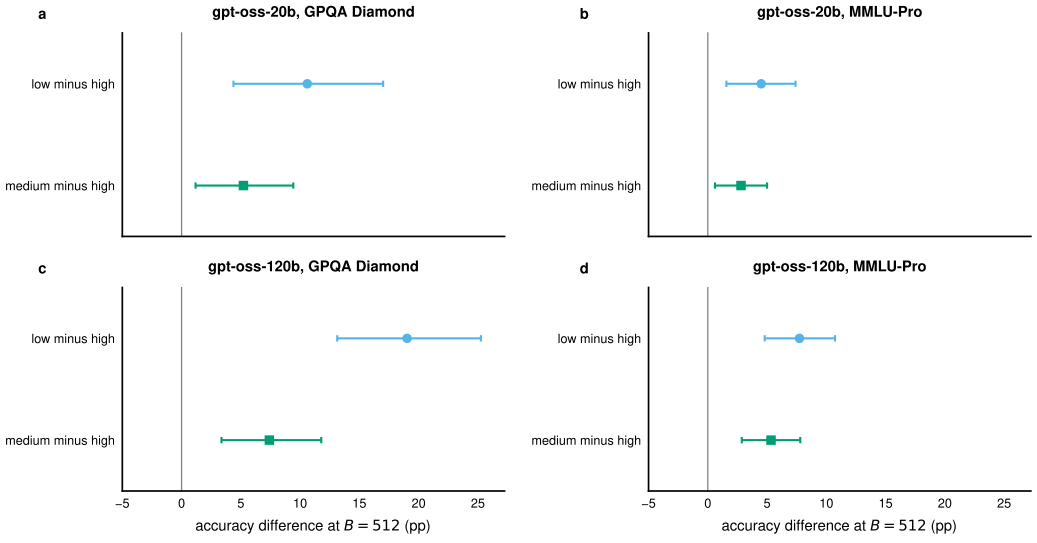


Figure A9: **Lower gpt-oss effort remains more accurate at the same 512-token checkpoint.** Terminal-aware candidate-logit accuracy differences compare low or medium effort with high effort for gpt-oss-20b (**a,b**) and gpt-oss-120b (**c,d**) on GPQA Diamond and MMLU-Pro. Naturally stopped traces retain their final answer. The 95% intervals average the three paired replicate outcomes within item and bootstrap items. Positive values favor the lower-effort setting.

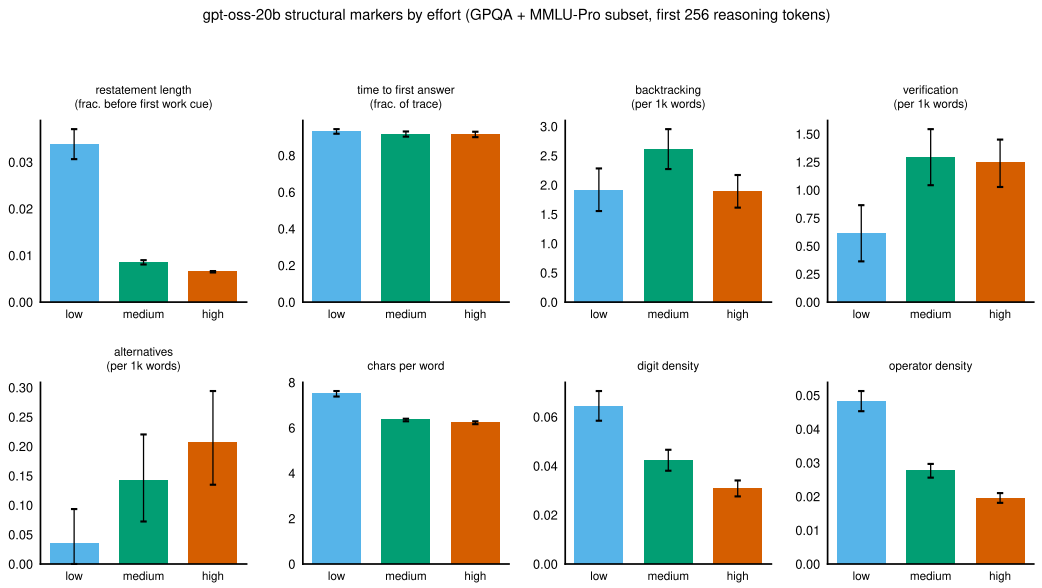


Figure A10: **Structural markers of the first** 256 **reasoning tokens by gpt-oss-20b effort setting.** Each panel is one length-normalized marker, averaged over GPQA Diamond and the stratified MMLU-Pro subset with 95% bootstrap intervals over items, for low (blue), medium (green), and high (orange) effort. Digit density, operator density, and characters per word decrease monotonically from low to high effort, while the explicit alternative-strategy marker rate increases. Together, these surface measures show that shorter-effort traces allocate more of their opening tokens to calculation-like text and contain fewer explicit alternatives. The time-to-first-answer marker is near-saturated, and neither position-based marker validates against GPT-5 mini annotations; the backtracking cue rate agrees with the annotations but is not monotone across effort, and the verification marker does not validate in the agreement analysis on this subset. We interpret the density and alternative-strategy markers, which are objective or annotation-supported (Section A).

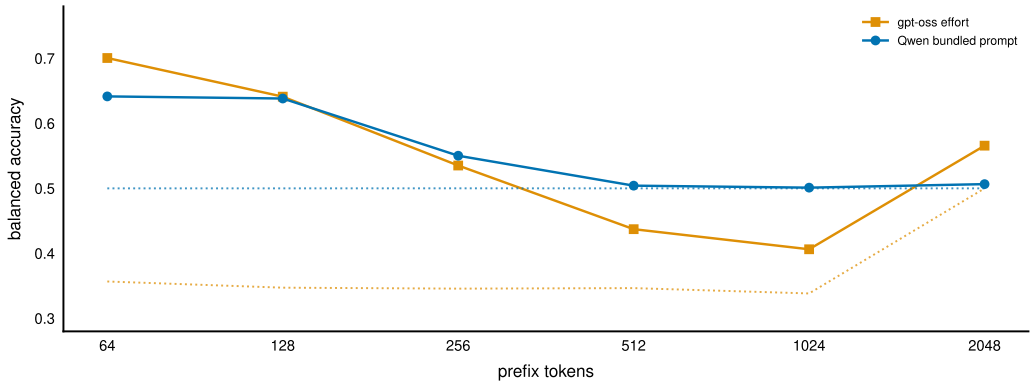


Figure A11: **Condition classifiers distinguish gpt-oss effort more strongly than Qwen bundled prompting.** Balanced accuracy of a classifier predicting the condition from the first k reasoning tokens, for gpt-oss effort (three-class) and Qwen3-14B bundled-prompt versus surprise-stop (two-class), with dotted label-shuffle baselines. Traces shorter than k are excluded at that prefix length, so the classifier is a measure of how well conditions can be distinguished among surviving traces rather than a causal estimate. The effort setting is readily distinguishable from the opening tokens; the bundled-prompt signal is weaker and decays to chance by 512 tokens.

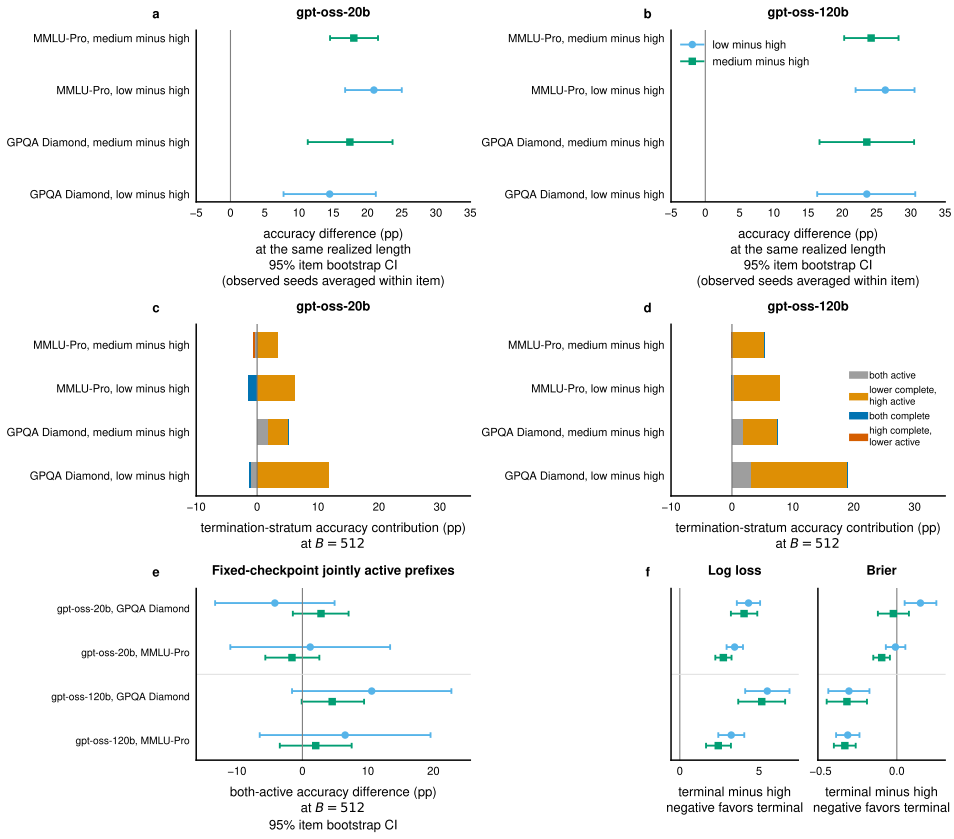


Figure A12: **Lower-effort accuracy gains concentrate in already-completed runs, while proper scores qualify the gain.** Candidate-logit analyses pair three replicates on GPQA Diamond and MMLU-Pro. **a,b**, Completed low- or medium-effort blocks are compared with reconstructable high-effort prefixes at the same itemwise horizon under a common medium-effort answer context; intervals average replicates within item and bootstrap items. **c,d**, Signed contributions of the $B = 512$ termination strata. **e**, Accuracy differences when both traces remain active at $B = 512$. **f**, Conditional log-loss and Brier-score differences at the matched horizon; negative values favor lower effort. Full accuracy, continuation, stratum, and score results are in Tables A22 to A24; Figure A8 shows completed-trace accuracy against median length.

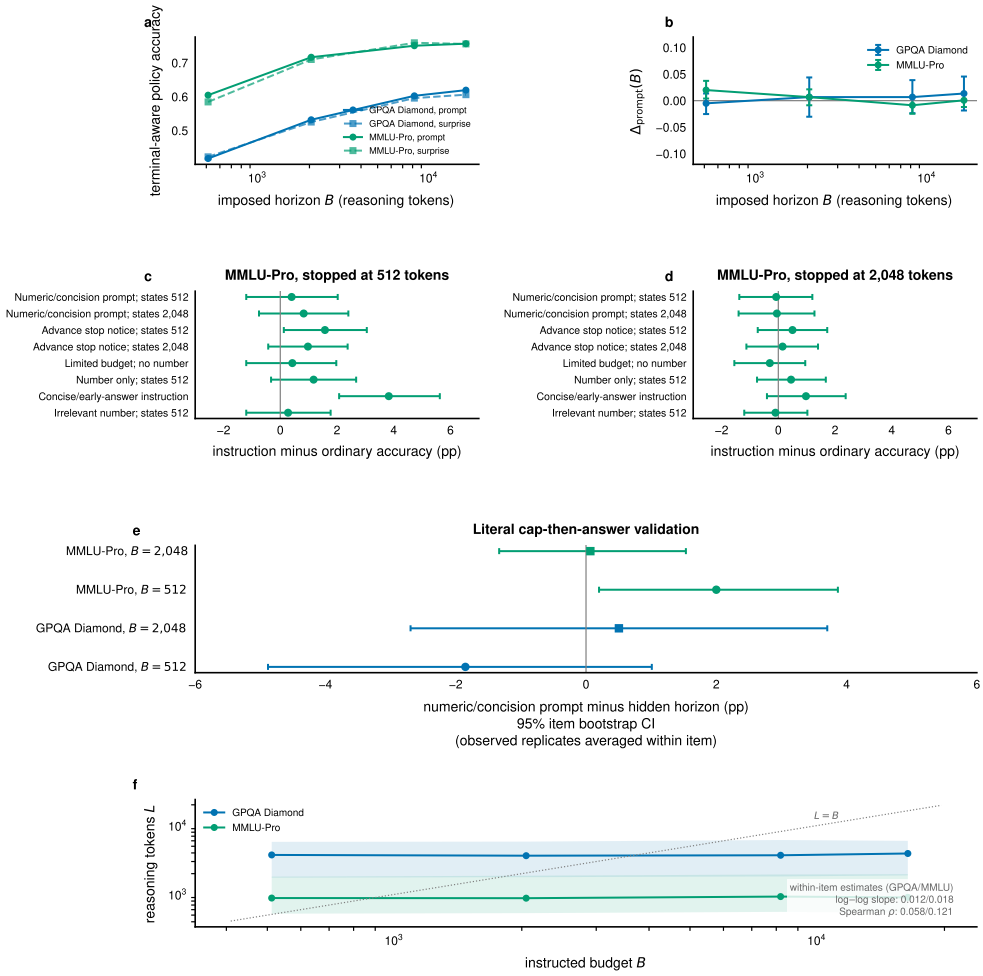


Figure A13: **Concision improves early answers, while the numeric/concision prompt gives smaller, mixed gains.** Token-exact analyses pair three replicates on GPQA Diamond (n=198) and MMLU-Pro (n=500) in panels **a,b,e,f**, and eight replicates on the same 500 MMLU-Pro items in panels **c,d**. Panels **a–d** use candidate-logit readouts. **a,b**, Terminal-aware accuracy under the numeric/concision prompt and surprise stop, and their difference; intervals average replicates within item and bootstrap items. **c,d**, Instruction-minus-ordinary accuracy at 512 and 2,048 tokens in the eight-replicate prompt-selection cohort. **e**, Literal cap-then-answer accuracy differences reproduce the benchmark split. **f**, Median reasoning length changes little with the announced budget; within-item log–log slopes are 0.012 for GPQA Diamond and 0.018 for MMLU-Pro. Tables A13 to A16 and A18 report the underlying estimates and sensitivity analyses.

D APPENDIX TABLES

The appendix tables report the estimates underlying the main comparisons and their diagnostics. Tables A8 to A11 define the estimands and interventions and validate the fixed-stop readouts, while Tables A13 to A24 report the Qwen and gpt-oss estimates and sensitivity analyses. Tables A26 to A30 give dense trajectory, termination-stratum, answer-context, arm-level, and score-tail detail; Tables A12 and A31 report single-rollout comparisons for Qwen3-8B and Omni-MATH-2 and characterize sampled trajectories. Qwen subset-replication and prompt-control results from cohorts analysed from decoded text are in Tables A32 to A34; readout validations, wrong-prefix continuation, activation readouts, and the prompt–effort interaction are in Tables A35, A36 and A42 to A44. Tables A37 to A39 report the completion-timing and parse-matched analyses of the Qwen generated-answer panel and its replication on questions not used to choose the prompts. These tables broaden item or prompt coverage and address measurement or diagnostic questions; the main matched-horizon claims rest on the token-exact experiments above.

Table A1: **The closest studies ask complementary questions about reasoning under limited computation.** We compare the scientific questions and evaluation targets, and state what this study adds relative to each.

Study	Scientific question	Evaluation target	What this study adds
Sun et al. (2025b)	Which models and prompts work under strict output budgets?	Accuracy subject to output limits and budget-dependent choices.	Completion-state decomposition of the fixed-allowance difference, paired effort and prompt interventions, and the announced-budget matching test.
Wu et al. (2025)	How robust is reasoning to interruption and finalization?	Answer quality and behavior when reasoning is interrupted.	Shared-allowance comparisons of paired interventions, with a completion-state decomposition of the differences.
Datta et al. (2026)	When do intermediate predictions stabilize?	Forced-answer predictions during reasoning and implications for early stopping.	Forced answers paired across interventions at shared allowances and checked against generated answers.
McFadyen et al. (2026)	How does inference-compute allocation affect model comparisons?	Performance across shared inference-compute ranges.	Shared reasoning allowances with a completion-state decomposition of the difference.
Ballon et al. (2026b)	How do answer probabilities evolve along reasoning trajectories?	Prefix-based answer-token distributions and trajectory diagnostics.	Paired interventions, generated-answer validation of the probe, completion stratification and announced-budget matching.
This study	What does a shorter-reasoning comparison establish?	Policy comparisons under natural completion, imposed horizons, and policy-chosen horizons, with completion-state and readout diagnostics.	—

Table A2: **The lower-complete/high-active stratum remains the largest positive contributor under alternative replicate pairings.** At $B = 512$, each of 10,000 draws independently permutes the three high-effort replicates within each item. The endpoint retains natural answers for completed runs and uses the candidate-logit readout for active prefixes. Contributions are percentage points of the unconditional accuracy difference on the stated balanced cohort. Minima and maxima are sensitivity ranges over the sampled re-pairings, not confidence intervals. Missing candidate-logit outcomes require excluding 0–5 items, depending on the comparison. Failures remain included in the generated-answer evaluation.

Model / task	Effort	Items	Excluded	Overall Δ	Original contribution	Re-pairing range	Largest (%)
120b / GPQA	low	198	0	$+19.0$	$+15.8$	$[+14.8, +16.5]$	100
120b / GPQA	medium	198	0	$+7.4$	$+5.6$	$[+5.1, +6.2]$	100
120b / MMLU-Pro	low	499	1	$+7.5$	$+7.3$	$[+7.1, +7.8]$	100
120b / MMLU-Pro	medium	500	0	$+5.3$	$+5.3$	$[+5.0, +5.9]$	100
20b / GPQA	low	198	0	$+10.6$	$+11.8$	$[+10.1, +12.5]$	100
20b / GPQA	medium	198	0	$+5.2$	$+3.4$	$[+2.9, +4.2]$	100
20b / MMLU-Pro	low	495	5	$+4.8$	$+6.4$	$[+5.8, +7.0]$	100
20b / MMLU-Pro	medium	499	1	$+2.8$	$+3.4$	$[+2.7, +4.2]$	100

Table A3: **Sampling and measurement procedures determine which comparisons support the main claims.** Cohorts differ in their sampling and measurement procedures. A token-exact intervention analysis, a replication on questions not used to choose the prompts, a calibration check, and a text-reconstructed replication provide different kinds of evidence.

Cohort	Sample and measurement	Role and limitation
Qwen3-14B and gpt-oss-20b/-120b source generations	GPQA Diamond: 198 items; MMLU-Pro: 500 items. Three stored generation replicates with token IDs and recorded seeds.	Source trajectories for intervention contrasts. The shared items and independent generation replicates are preserved across horizons.
gpt-oss shared-budget continuations	Same 12,564 stored gpt-oss source trajectories; 87,948 generated-answer outcomes at seven horizons, with $R = 16$.	Main fixed-allowance result. Exact completion-token prefixes; source-policy prompts reconstructed with a fixed template version and date. All outcomes included and channel boundaries checked.
Qwen3-14B nine-arm prompt experiment	Same 500 MMLU-Pro items; eight replicates. Exact prefix readouts at 512 and 2,048 tokens. Six instructions have 96,000 generated $R = 64$ answers across source-policy and common contexts.	Prompt contrasts and announced-budget interaction. Prompt selection used this cohort, so it does not test generalization to new questions.
Qwen3-14B replication cohort	Fresh category-stratified 500 MMLU-Pro items disjoint from both earlier subsets; eight independently seeded generations; six instructions; 96,000 generated $R = 64$ answers, plus four concise-plus-number instructions generated afterwards on the same items and seeds (64,000 answers).	Token-exact replication on questions not used to choose the prompts. One model and one benchmark.
Fixed-stop calibration and terminal replay checks	Separate calibration cohorts, including the 2,788-prefix gpt-oss terminal-boundary check.	Readout validation. A check on these prefixes does not certify every trajectory in the main cohorts.
Second MMLU-Pro subset and prompt controls	Records analysed from decoded text rather than stored token IDs.	Descriptive replication and scope checks; excluded from main token-exact estimates.
Omni-MATH-2 comparisons	Single rollouts: Qwen 200 items; gpt-oss-20b 475 and -120b 490 available paired items.	Open-ended scope of the observed comparisons; no estimate of variation across independent generations.
Within-item replicate re-pairing	Candidate-logit fixed-512 gpt-oss outcomes, restricted separately to items with all three outcomes under both compared efforts.	Pairing sensitivity analysis. Its balanced population and re-pairing ranges are reported in Table A2.
Generated-answer re-pairing and answer-allowance check	Full seven-horizon cohort for re-pairing; same seeded 100-item subset per task for $R = 16$ versus $R = 64$.	No outcome exclusions. Pairing ranges are not sampling intervals; the allowance check also varies replay batching (Tables A6 and A7).

Table A4: **Lower-effort advantages at a short allowance can reverse at a longer one.** Source-policy cap-then-answer accuracy differences from high effort, in percentage points, on all 198 GPQA Diamond or 500 MMLU-Pro items with three independent source generations per effort. Completed runs retain their original answer; active runs receive a greedy answer continuation with $R = 16$. Every planned outcome is retained, with parsing failures incorrect. Pointwise and simultaneous 95% intervals use 10,000 paired item-bootstrap draws, averaging replicates within item. Each simultaneous band covers both contrasts and all seven horizons within one model–benchmark comparison. These reused cohorts support descriptive comparisons, not an estimate of selecting a policy on separate questions.

Model / task	Effort	Δ_{512}	Pointwise CI	Simultaneous CI	Δ_{8192}	Pointwise CI	Simultaneous CI
20b / GPQA	low	+11.1	[+4.7, +17.5]	[+3.7, +18.5]	−6.4	[−11.6, −1.3]	[−13.8, +1.0]
20b / GPQA	medium	+4.7	[+0.3, +9.1]	[−2.7, +12.1]	+1.9	[−2.2, +5.9]	[−5.6, +9.3]
20b / MMLU-Pro	low	+4.5	[+1.5, +7.7]	[+0.6, +8.5]	−8.7	[−11.2, −6.1]	[−12.6, −4.7]
20b / MMLU-Pro	medium	+3.1	[+0.9, +5.3]	[−0.9, +7.0]	−0.7	[−2.6, +1.1]	[−4.7, +3.2]
120b / GPQA	low	+18.9	[+12.8, +24.9]	[+11.4, +26.3]	−8.8	[−13.5, −4.2]	[−16.2, −1.3]
120b / GPQA	medium	+6.9	[+2.5, +11.3]	[−0.5, +14.3]	−5.2	[−9.4, −1.2]	[−12.6, +2.2]
120b / MMLU-Pro	low	+7.7	[+4.7, +10.8]	[+3.7, +11.8]	−4.4	[−6.5, −2.3]	[−8.5, −0.3]
120b / MMLU-Pro	medium	+5.2	[+2.7, +7.8]	[+1.1, +9.3]	−0.8	[−2.6, +1.0]	[−4.9, +3.3]

Table A5: **Mean token expenditure can differ substantially from typical trace length.** Natural final accuracy (%) and all generated tokens, including channel boundaries and the answer, for the same complete cohorts as Table A4. Each policy has 594 GPQA Diamond or 1,500 MMLU-Pro source generations. Runs reaching the original 126,000-token source limit remain in the token statistics; missing final answers count as incorrect. These are observed token costs within each model, not measurements of cross-model compute or online latency.

Model / task	Effort	Accuracy	Mean tokens	Median	90th percentile	Parse failures
20b / GPQA	low	55.9	497.7	294.0	1,137.1	0
20b / GPQA	medium	64.6	9,167.2	1,816.0	29,555.2	2
20b / GPQA	high	66.0	37,119.7	14,467.5	119,113.5	45
20b / MMLU-Pro	low	64.2	170.8	100.5	390.0	6
20b / MMLU-Pro	medium	72.3	1,116.0	496.5	2,400.0	2
20b / MMLU-Pro	high	74.5	7,792.4	1,413.5	23,087.2	7
120b / GPQA	low	65.0	338.8	264.5	689.2	0
120b / GPQA	medium	68.5	1,466.1	933.5	3,617.8	0
120b / GPQA	high	79.3	11,291.0	4,402.0	33,107.4	1
120b / MMLU-Pro	low	75.5	171.2	89.0	409.1	1
120b / MMLU-Pro	medium	79.1	517.3	266.5	1,162.2	0
120b / MMLU-Pro	high	79.8	3,669.4	972.5	8,964.1	0

Table A6: **The advantage remains concentrated in completed lower-effort runs after re-pairing generated answers on the full cohort.** At $B = 512$, the lower-complete/high-active stratum is the largest positive contributor in all 10,000 independent within-item re-pairings of the three high-effort replicates. Contributions and ranges are percentage points of the unconditional source-policy cap-then-answer accuracy difference with $R = 16$. No items are excluded. The overall difference remains invariant; ranges are pairing sensitivity, not sampling confidence intervals or causal stopping effects. The complete seven-horizon decomposition is provided in the analysis data.

Model / task	Effort	Items	Overall Δ	Original contribution	Re-pairing range	Largest (%)
20b / GPQA	low	198	+11.1	+12.1	[+10.4, +12.8]	100
20b / GPQA	medium	198	+4.7	+3.4	[+2.9, +4.4]	100
20b / MMLU-Pro	low	500	+4.5	+6.2	[+5.5, +6.7]	100
20b / MMLU-Pro	medium	500	+3.1	+3.3	[+2.7, +4.3]	100
120b / GPQA	low	198	+18.9	+16.3	[+15.5, +17.2]	100
120b / GPQA	medium	198	+6.9	+6.4	[+5.7, +6.9]	100
120b / MMLU-Pro	low	500	+7.7	+7.7	[+7.5, +8.3]	100
120b / MMLU-Pro	medium	500	+5.2	+5.3	[+5.0, +6.1]	100

Table A7: **A larger final-answer allowance changes each policy's accuracy by at most one percentage point in the sensitivity check.** The same seeded 100-item subset per benchmark is evaluated at $R = 16$ and $R = 64$, with three replicates and all seven reasoning allowances (2,100 outcomes per row). The accuracy column gives the range of changes across horizons, in percentage points; failure and exhaustion counts pool horizons. Naturally completed answers are retained identically. Greedy continuations are re-evaluated in different batches, so differences combine the allowance change with numerical sensitivity of the serving backend. An exhausted allowance need not be a parsing failure; every unparseable answer is incorrect.

Model / task	Effort	Accuracy change range	Failures $R16$	Failures $R64$	Exhausted $R16$	Exhausted $R64$
20b / GPQA	low	$[+0.0, +0.7]$	1	1	3	1
20b / GPQA	medium	$[+0.0, +0.7]$	2	3	6	2
20b / GPQA	high	$[-0.7, +1.0]$	3	3	7	3
20b / MMLU-Pro	low	$[-0.7, +0.0]$	13	13	0	0
20b / MMLU-Pro	medium	$[-0.3, +0.3]$	2	2	2	0
20b / MMLU-Pro	high	$[-1.0, +0.7]$	7	8	8	0
120b / GPQA	low	$[+0.0, +0.0]$	0	0	0	0
120b / GPQA	medium	$[-0.3, +0.3]$	1	1	0	0
120b / GPQA	high	$[-0.7, +0.3]$	3	4	0	0
120b / MMLU-Pro	low	$[+0.0, +0.0]$	0	0	0	0
120b / MMLU-Pro	medium	$[+0.0, +0.0]$	1	1	0	0
120b / MMLU-Pro	high	$[+0.0, +0.3]$	0	0	2	0

Table A8: **We evaluate each condition at a common stopping point and separate completion from prefix quality.** The terminal-aware readout retains a natural answer after completion and probes an unfinished prefix in its source context. Common-context replay applies the same answer context to both conditions; both-active comparisons further restrict the analysis to jointly unfinished pairs. Behavioral continuations generate an unconstrained short answer from the stopped prefix.

Comparison	Where the comparison stops	How the answer is measured	Main result
gpt-oss effort at shared allowances	All three efforts; seven shared allowances from 128 to 8,192 tokens	Source-policy cap-then-answer accuracy with $R = 16$; $\Delta_{z,z'}(B)$	Compares delivered answers on the full cohort under externally fixed allowances
Qwen six-instruction generated answers	Shared allowances $B \in \{512, 2048\}$ for all six instructions	Fresh source-policy answers with $R = 64$, including completed reasoning blocks. All outcomes retained.	Concision improves early answers. Matching the announced number adds little.
Qwen textual treatment: visible numeric/concision prompt (v) vs ordinary generation (h)	Common imposed horizon B; retain either arm's natural answer if it completed before B	Terminal-aware policy-context candidate-logit readout; $\tau_{Q,\mathrm{pol}}(B) = \mathbb{E}[Y^{v,T_{\mathrm{pol}}}(B) - Y^{h,T_{\mathrm{pol}}}(B)]$	Shorter traces; no broad accuracy gain at the same imposed horizon
Qwen prefix diagnostics: tested instruction (a) vs ordinary generation (h)	Common imposed horizon B; replay both prefixes under the ordinary base prompt	Correct completion by B, common-context terminal replay, both-active contrast, and four-stratum decomposition	Locates full-cohort differences and describes selected unfinished pairs
Qwen literal cap: visible numeric/concision prompt vs hidden horizon	Common external cap $B \in \{512, 2048\}$; both arms then receive an answer continuation	Greedy generated-answer continuation with $R = 16$; $\mathbb{E}[Y^{v,C}(B) - Y^{h,C}(B)]$	Benchmark split checked without option normalization
gpt-oss effort: completed $c \in \{\mathrm{low}, \mathrm{medium}\}$ vs high effort	Itemwise lower-effort terminal length L_{ir}^{c}; replay every reconstructable high prefix, including early completions	All-replayable common-context probe; $\tau_{G,R}^{c} = \mathbb{E}[D^{c} \mid R^{c} = 1]$, with worst-case bounds	Lower effort reaches correct terminal answers earlier
gpt-oss active diagnostic	The same matched pairs restricted to high effort still active at L_{ir}^{c}	Common-context candidate-logit probe	Separates the policy-cost result from early high-effort completion
gpt-oss continuation check: the same matched prefixes	The same all-replayable L_{ir}^{c} pairs	Greedy unconstrained final-channel answer; $\mathbb{E}[Y^{c,C} - Y^{\mathrm{high},C} \mid R^{c} = 1]$	Tests whether the ordering survives a generated-answer continuation

Table A9: **Verbatim generation and readout interventions.** Braces indicate the interpolated token budget.

Condition	Message role	Exact text or template
Qwen numeric/concision prompt	system, appended to base instruction	`You have at most {B} tokens inside the <think>...</think> reasoning block. Use a strategy appropriate for this budget. When you have enough evidence, close the reasoning block and answer with only the final answer.`
Qwen limited-budget control	system, appended to base instruction	`You have a limited token budget inside the <think>...</think> reasoning block. Use a strategy appropriate for this limit. When you have enough evidence, close the reasoning block and answer with only the final answer.`
Qwen number-only control	system, appended to base instruction	`You have at most {B} tokens inside the <think>...</think> reasoning block.`
Qwen advance exact-stop control	system, appended to base instruction	`Your <think>...</think> reasoning block will be stopped after exactly {B} tokens.`
Qwen approximate-budget control	system, appended to base instruction	`Aim to use approximately {B} tokens inside the <think>...</think> reasoning block.`
Qwen irrelevant-number control	system, appended to base instruction	`The control number for this problem is {B}. Do not treat this number as a token budget.`
Qwen concise/early-answer control	system, appended to base instruction	`Use a very concise reasoning strategy. Prioritize reaching a usable answer quickly.`
Qwen thorough semantic control	system, appended to base instruction	`Use a thorough reasoning strategy. Explore alternatives and check your answer carefully before finalizing.`
Qwen limited-time closure	continuation	`Considering the limited time by the user, I have to give the solution based on the thinking directly now. </think>`
Qwen paraphrased closure	continuation	`Based on my reasoning so far, my answer is: </think>`
Qwen neutral closure	continuation	`</think>`
Qwen budget-present readout	system, answer-time only	`You have at most {B} tokens inside the <think>...</think> reasoning block. Use a strategy appropriate for this budget. When you have enough evidence, close the reasoning block and answer with only the final answer.`
Qwen answer instruction	system	`Solve the following problem. Please make sure that your response only consists of a single letter corresponding to the correct answer choice. Do not include anything else in your final response.`
gpt-oss effort	chat-template kwarg/system scaffold	`reasoning_effort = low | medium | high`, rendered by the snapshot chat template as `Reasoning: low|medium|high` in the system message before the developer/user messages

Table A10: **Fixed-stop calibration separates completed answers from the selected set of surviving traces.** The full-vocabulary candidate-logit readout is evaluated at $B \in \{0, 512, 2{,}048, 8{,}192, 16{,}384\}$. Survival is $P(L > B)$, survivor-only accuracy conditions on $L > B$, and terminal-aware accuracy retains the natural answer after completion. The natural full-trace reference uses the same cohort. Rate columns report percentages with 95% item-bootstrap intervals. Accuracy omits missing parsed outcomes; n retains the full cohort.

Model	Benchmark	B	n	Survival (%)	Survivor-only accuracy (%)	Terminal-aware policy accuracy (%)	Natural full-trace accuracy (%)
Qwen3-4B	GPQA Diamond	0	198	100.0 [100.0, 100.0]	31.8 [25.3, 38.4]	31.8 [25.3, 38.4]	55.6 [48.5, 62.2]
Qwen3-4B	GPQA Diamond	512	198	100.0 [100.0, 100.0]	34.8 [28.3, 41.4]	34.8 [28.3, 41.4]	55.6 [48.5, 62.2]
Qwen3-4B	GPQA Diamond	2,048	198	85.9 [80.8, 90.4]	37.6 [30.6, 44.7]	43.4 [36.9, 50.5]	55.6 [48.5, 62.2]
Qwen3-4B	GPQA Diamond	8,192	198	23.2 [17.7, 29.3]	50.0 [34.8, 65.2]	56.1 [49.5, 62.6]	55.6 [48.5, 62.2]
Qwen3-4B	GPQA Diamond	16,384	198	2.5 [0.5, 5.1]	40.0 [0.0, 80.0]	54.8 [47.7, 61.9]	55.6 [48.5, 62.2]
Qwen3-4B	MMLU-Pro	0	500	100.0 [100.0, 100.0]	39.0 [34.8, 43.2]	39.0 [34.8, 43.2]	69.5 [65.5, 73.5]
Qwen3-4B	MMLU-Pro	512	500	85.4 [82.2, 88.4]	43.3 [38.9, 48.0]	49.4 [45.2, 53.8]	69.5 [65.5, 73.5]
Qwen3-4B	MMLU-Pro	2,048	500	35.2 [31.0, 39.4]	44.3 [36.9, 51.7]	64.3 [60.1, 68.5]	69.5 [65.5, 73.5]
Qwen3-4B	MMLU-Pro	8,192	500	8.0 [5.6, 10.4]	27.5 [15.0, 42.5]	68.3 [64.3, 72.3]	69.5 [65.5, 73.5]
Qwen3-4B	MMLU-Pro	16,384	500	1.0 [0.2, 2.0]	0.0 [0.0, 0.0]	69.3 [65.3, 73.3]	69.5 [65.5, 73.5]
Qwen3-8B	GPQA Diamond	0	198	100.0 [100.0, 100.0]	30.8 [24.2, 37.4]	30.8 [24.2, 37.4]	57.9 [50.8, 64.5]
Qwen3-8B	GPQA Diamond	512	198	98.0 [96.0, 99.5]	35.6 [28.9, 42.3]	36.9 [30.3, 43.4]	57.9 [50.8, 64.5]
Qwen3-8B	GPQA Diamond	2,048	198	86.4 [81.3, 90.9]	35.7 [28.7, 42.7]	42.9 [35.9, 49.5]	57.9 [50.8, 64.5]
Qwen3-8B	GPQA Diamond	8,192	198	27.8 [21.7, 34.3]	30.9 [18.2, 43.6]	56.9 [49.7, 63.5]	57.9 [50.8, 64.5]
Qwen3-8B	GPQA Diamond	16,384	198	4.0 [1.5, 7.1]	37.5 [0.0, 75.0]	57.9 [50.8, 64.5]	57.9 [50.8, 64.5]
Qwen3-8B	MMLU-Pro	0	500	100.0 [100.0, 100.0]	43.0 [38.8, 47.4]	43.0 [38.8, 47.4]	72.2 [68.1, 76.0]
Qwen3-8B	MMLU-Pro	512	500	86.8 [83.8, 89.8]	46.5 [41.7, 51.2]	52.6 [48.2, 57.0]	72.2 [68.1, 76.0]
Qwen3-8B	MMLU-Pro	2,048	500	43.0 [38.8, 47.4]	43.7 [37.2, 50.2]	65.1 [60.9, 69.3]	72.2 [68.1, 76.0]
Qwen3-8B	MMLU-Pro	8,192	500	10.0 [7.4, 12.8]	34.0 [22.0, 48.0]	70.2 [66.2, 74.2]	72.2 [68.1, 76.0]
Qwen3-8B	MMLU-Pro	16,384	500	1.8 [0.8, 3.0]	33.3 [0.0, 66.7]	71.8 [67.7, 75.8]	72.2 [68.1, 76.0]
Qwen3-14B	GPQA Diamond	0	198	100.0 [100.0, 100.0]	37.4 [30.8, 43.9]	37.4 [30.8, 43.9]	62.6 [55.6, 69.2]
Qwen3-14B	GPQA Diamond	512	198	99.0 [97.5, 100.0]	41.8 [35.2, 49.0]	42.4 [35.8, 49.5]	62.6 [55.6, 69.2]
Qwen3-14B	GPQA Diamond	2,048	198	76.8 [70.7, 82.3]	44.1 [36.2, 52.0]	53.5 [46.5, 60.6]	62.6 [55.6, 69.2]
Qwen3-14B	GPQA Diamond	8,192	198	18.7 [13.6, 24.2]	29.7 [16.2, 45.9]	59.1 [52.5, 66.2]	62.6 [55.6, 69.2]
Qwen3-14B	GPQA Diamond	16,384	198	0.0 [0.0, 0.0]	—	62.6 [56.1, 69.2]	62.6 [55.6, 69.2]
Qwen3-14B	MMLU-Pro	0	500	100.0 [100.0, 100.0]	50.4 [46.0, 54.8]	50.4 [46.0, 54.8]	76.8 [73.0, 80.4]
Qwen3-14B	MMLU-Pro	512	500	80.6 [77.0, 84.0]	52.4 [47.4, 57.1]	60.0 [55.8, 64.4]	76.8 [73.0, 80.4]
Qwen3-14B	MMLU-Pro	2,048	500	31.0 [27.0, 35.0]	50.3 [42.6, 58.1]	70.8 [66.8, 74.8]	76.8 [73.0, 80.4]
Qwen3-14B	MMLU-Pro	8,192	500	5.8 [3.8, 8.0]	51.7 [34.5, 69.0]	76.6 [72.8, 80.2]	76.8 [73.0, 80.4]
Qwen3-14B	MMLU-Pro	16,384	500	1.0 [0.2, 2.0]	0.0 [0.0, 0.0]	76.4 [72.8, 80.2]	76.8 [73.0, 80.4]
gpt-oss-20b	GPQA Diamond	0	198	100.0 [100.0, 100.0]	40.9 [33.8, 48.0]	40.9 [33.8, 48.0]	62.6 [55.6, 69.2]
gpt-oss-20b	GPQA Diamond	512	198	83.8 [78.3, 88.9]	42.2 [34.3, 49.4]	50.0 [42.9, 57.1]	62.6 [55.6, 69.2]
gpt-oss-20b	GPQA Diamond	2,048	198	43.9 [37.4, 51.0]	36.8 [26.4, 47.1]	60.6 [53.5, 67.2]	62.6 [55.6, 69.2]
gpt-oss-20b	GPQA Diamond	8,192	198	9.1 [5.6, 13.1]	38.9 [16.7, 61.1]	63.6 [56.6, 70.2]	62.6 [55.6, 69.2]
gpt-oss-20b	GPQA Diamond	16,384	198	2.5 [0.5, 5.1]	60.0 [20.0, 100.0]	63.6 [57.1, 70.2]	62.6 [55.6, 69.2]
gpt-oss-20b	MMLU-Pro	0	500	100.0 [100.0, 100.0]	43.4 [39.2, 47.8]	43.4 [39.2, 47.8]	72.9 [68.9, 76.8]
gpt-oss-20b	MMLU-Pro	512	500	40.0 [35.8, 44.2]	35.5 [29.0, 42.0]	64.1 [59.9, 68.3]	72.9 [68.9, 76.8]
gpt-oss-20b	MMLU-Pro	2,048	500	10.0 [7.4, 12.6]	24.0 [12.0, 36.0]	71.1 [67.1, 75.2]	72.9 [68.9, 76.8]
gpt-oss-20b	MMLU-Pro	8,192	500	0.4 [0.0, 1.0]	0.0 [0.0, 0.0]	72.9 [69.1, 76.8]	72.9 [68.9, 76.8]
gpt-oss-20b	MMLU-Pro	16,384	500	0.0 [0.0, 0.0]	—	72.9 [69.1, 76.8]	72.9 [68.9, 76.8]
gpt-oss-120b	GPQA Diamond	0	198	100.0 [100.0, 100.0]	38.9 [32.3, 46.0]	38.9 [32.3, 46.0]	69.7 [63.1, 75.8]
gpt-oss-120b	GPQA Diamond	512	198	70.2 [63.6, 76.8]	36.0 [28.1, 43.9]	51.5 [44.4, 58.6]	69.7 [63.1, 75.8]
gpt-oss-120b	GPQA Diamond	2,048	198	23.7 [18.2, 29.8]	40.4 [25.5, 53.2]	67.7 [61.1, 74.2]	69.7 [63.1, 75.8]
gpt-oss-120b	GPQA Diamond	8,192	198	0.5 [0.0, 1.5]	0.0 [0.0, 0.0]	69.7 [63.1, 75.8]	69.7 [63.1, 75.8]
gpt-oss-120b	GPQA Diamond	16,384	198	0.0 [0.0, 0.0]	—	69.7 [63.1, 75.8]	69.7 [63.1, 75.8]
gpt-oss-120b	MMLU-Pro	0	500	100.0 [100.0, 100.0]	48.4 [44.0, 52.8]	48.4 [44.0, 52.8]	78.4 [74.8, 81.8]
gpt-oss-120b	MMLU-Pro	512	500	27.2 [23.4, 31.2]	40.4 [32.4, 48.5]	71.2 [67.2, 75.2]	78.4 [74.8, 81.8]
gpt-oss-120b	MMLU-Pro	2,048	500	4.2 [2.6, 6.0]	42.9 [23.8, 66.7]	78.0 [74.4, 81.6]	78.4 [74.8, 81.8]
gpt-oss-120b	MMLU-Pro	8,192	500	0.0 [0.0, 0.0]	—	78.4 [74.8, 82.0]	78.4 [74.8, 81.8]
gpt-oss-120b	MMLU-Pro	16,384	500	0.0 [0.0, 0.0]	—	78.4 [74.8, 82.0]	78.4 [74.8, 81.8]

Table A11: **Completed-prefix readouts agree with natural answers in** 98.3**–**100.0% **of cases.** Each row compares the candidate-logit answer from a replayed completed reasoning prefix with the parsed natural answer from the same trace. This separate cohort validates terminal replay; it does not assess coverage of the three-replicate token-exact analysis.

Model	Benchmark	Source policy	Compared	Agree	Agreement
gpt-oss-120b	GPQA Diamond	gpt-oss low effort	198	198	100.0%
gpt-oss-120b	GPQA Diamond	gpt-oss medium effort	198	198	100.0%
gpt-oss-120b	MMLU-Pro	gpt-oss low effort	499	499	100.0%
gpt-oss-120b	MMLU-Pro	gpt-oss medium effort	500	500	100.0%
gpt-oss-20b	GPQA Diamond	gpt-oss low effort	196	196	100.0%
gpt-oss-20b	GPQA Diamond	gpt-oss medium effort	196	196	100.0%
gpt-oss-20b	MMLU-Pro	gpt-oss low effort	496	494	99.6%
gpt-oss-20b	MMLU-Pro	gpt-oss medium effort	499	499	100.0%
Qwen3-14B	GPQA Diamond	Qwen numeric/concision prompt	1,317	1,315	99.8%
Qwen3-14B	GPQA Diamond	Qwen surprise	588	586	99.7%
Qwen3-14B	MMLU-Pro	Qwen numeric/concision prompt	4,451	4,447	99.9%
Qwen3-14B	MMLU-Pro	Qwen surprise	1,489	1,463	98.3%

Table A12: **Open-ended continuations show no consistent Qwen prompt gain and a gpt-oss matched-horizon advantage.** GPT-5 mini judges free-form Omni-MATH-2 answers. Qwen rows compare numeric/concision-prompt and surprise-stop prefixes at the same imposed horizon on a 200-item sample; each row includes only item–horizon comparisons for which both answers were judged. gpt-oss rows compare a completed lower-effort reasoning block with high effort at the itemwise lower-effort terminal horizon, replaying a completed high-effort reasoning block when it finishes before that horizon. Every reasoning block is read under a common medium-effort final-answer context. The one-rollout, difficulty-stratified 500-item draw yields model-specific paired cohorts of 475 items for gpt-oss-20b and 490 for gpt-oss-120b. The analysis includes all 15,440 judged gpt-oss answers across the matched-horizon and fixed-checkpoint comparisons. Intervals bootstrap items.

Comparison	Model	Contrast	Stop	n	Treated/lower acc.	Comparator acc.	Δ [95% CI]
Qwen prompt	Qwen3-14B	prompt minus surprise	512	170	22.9%	19.4%	+3.5 [−1.2, +8.2]
Qwen prompt	Qwen3-14B	prompt minus surprise	2,048	150	33.3%	31.3%	+2.0 [−3.3, +7.3]
Qwen prompt	Qwen3-14B	prompt minus surprise	8,192	73	26.0%	30.1%	−4.1 [−12.3, +4.1]
gpt-oss effort	gpt-oss-20b	low minus high prefix	matched	475	47.8%	14.3%	+33.5 [+28.8, +37.9]
gpt-oss effort	gpt-oss-20b	medium minus high prefix	matched	475	61.5%	27.6%	+33.9 [+28.8, +38.9]
gpt-oss effort	gpt-oss-120b	low minus high prefix	matched	490	61.2%	19.2%	+42.0 [+37.3, +46.7]
gpt-oss effort	gpt-oss-120b	medium minus high prefix	matched	490	70.6%	33.1%	+37.6 [+32.4, +42.7]

Table A13: **The numeric/concision prompt does not consistently improve Qwen accuracy at the same horizon.** Panel A compares the numeric/concision (visible-horizon) and ordinary (hidden-horizon) conditions at the same imposed horizon using token-exact replay and the terminal-aware policy-context candidate-logit readout for all 198 GPQA Diamond and 500 MMLU-Pro items across three generation replicates; n gives pointwise/complete-curve item counts. Δ is visible minus hidden accuracy; intervals and simultaneous bands average replicates within item before bootstrapping items. AUC integrates Δ over $\log_2 B$. Panel B reports the separate literal cap-then-answer validation with $R = 16$, with parse coverage shown as visible/hidden.

Panel A: terminal-aware policy readout									
Model	Benchmark	B	n	Replicates	Visible acc.	Hidden acc.	Δ [95% CI]	Simultaneous 95% band	AUC [95% CI]
Qwen3-14B	GPQA Diamond	512	198/198	3	41.8%	42.3%	−0.5 [−2.5, +1.3]	[−4.7, +3.7]	+0.5 [−1.6, +2.7]
Qwen3-14B	GPQA Diamond	2,048	198/198	3	53.2%	52.5%	+0.7 [−3.0, +4.4]	[−3.5, +4.9]	—
Qwen3-14B	GPQA Diamond	8,192	198/198	3	60.3%	59.6%	+0.7 [−2.4, +3.9]	[−3.5, +4.9]	—
Qwen3-14B	GPQA Diamond	16,384	198/198	3	62.0%	60.6%	+1.3 [−1.9, +4.5]	[−2.9, +5.6]	—
Qwen3-14B	MMLU-Pro	512	500/500	3	60.5%	58.5%	+2.0 [+0.4, +3.7]	[+0.1, +3.9]	+0.4 [−0.7, +1.5]
Qwen3-14B	MMLU-Pro	2,048	500/500	3	71.7%	71.0%	+0.7 [−0.9, +2.1]	[−1.3, +2.6]	—
Qwen3-14B	MMLU-Pro	8,192	500/500	3	75.1%	75.9%	−0.9 [−2.5, +0.7]	[−2.8, +1.1]	—
Qwen3-14B	MMLU-Pro	16,384	500/500	3	75.7%	75.6%	+0.1 [−1.2, +1.3]	[−1.9, +2.0]	—
Panel B: literal cap-then-answer continuation									
Model	Benchmark	B	n	Replicates	Visible acc.	Hidden acc.	Δ [95% CI]	Parse coverage V/H	Policy-readout Δ
Qwen3-14B	GPQA Diamond	512	198	3	39.9%	41.8%	−1.9 [−4.9, +1.0]	100.0%/100.0%	−0.5
Qwen3-14B	GPQA Diamond	2,048	198	3	53.9%	53.4%	+0.5 [−2.7, +3.7]	99.8%/99.8%	+0.7
Qwen3-14B	MMLU-Pro	512	500	3	60.7%	58.7%	+2.0 [+0.2, +3.9]	97.1%/96.7%	+2.0
Qwen3-14B	MMLU-Pro	2,048	500	3	71.4%	71.3%	+0.1 [−1.3, +1.5]	99.5%/97.5%	+0.7

Table A14: **Concision yields the largest 512-token candidate-logit gain, with no clear matching-number benefit.** **a**, Qwen3-14B results on the 500-item MMLU-Pro subset with eight paired replicates per item. "Finished" is the share of runs that produce a natural answer by B; otherwise, accuracy uses the exact stored prefix and candidate-logit readout. Differences are relative to no advance notice, with nested item–replicate bootstrap intervals. **b**, Direct tests of whether matching the stated number to the imposed stop helps.

a Nine prompt instructions

			Imposed stop $B = 512$			Imposed stop $B = 2{,}048$		
Prompt instruction	Number stated H	Median length	Finished	Accuracy	Difference [95% CI]	Finished	Accuracy	Difference [95% CI]
No advance notice	—	1,083	20.3%	59.7%	—	72.1%	72.1%	—
Numeric/concision prompt	512	900	25.0%	60.1%	+0.4 [−1.2, +2.0]	77.6%	72.0%	−0.1 [−1.4, +1.2]
Numeric/concision prompt	2,048	921	23.5%	60.5%	+0.8 [−0.8, +2.4]	77.1%	72.1%	−0.1 [−1.4, +1.3]
Advance notice of exact stop	512	857	25.4%	61.3%	+1.6 [+0.1, +3.1]	78.4%	72.6%	+0.5 [−0.7, +1.7]
Advance notice of exact stop	2,048	900	23.0%	60.7%	+1.0 [−0.4, +2.4]	77.3%	72.3%	+0.2 [−1.1, +1.4]
Limited budget (no number)	—	874	25.0%	60.1%	+0.4 [−1.2, +2.0]	77.1%	71.8%	−0.3 [−1.6, +1.0]
Number only	512	924	24.1%	60.9%	+1.2 [−0.3, +2.7]	77.1%	72.6%	+0.5 [−0.8, +1.7]
Concise/early-answer instruction	—	679	36.4%	63.5%	+3.8 [+2.1, +5.6]	82.3%	73.1%	+1.0 [−0.4, +2.4]
Irrelevant number	512	1,020	21.2%	60.0%	+0.3 [−1.2, +1.8]	73.0%	72.0%	−0.1 [−1.2, +1.0]

b Direct test of matching the stated number to the stop

Prompt instruction	Match advantage (pp)	95% CI
Numeric/concision prompt	−0.2	[−1.0, +0.6]
Advance notice of exact stop	+0.1	[−0.6, +0.8]

Table A15: **At $B = 512$, the candidate-logit concise/early-answer contrast combines earlier completion with better unfinished prefixes. a**, Correct completion, all-pair accuracy, and both-active accuracy for concise/early-answer versus ordinary generation. Both-active rows condition on pairs in which neither trace has finished. **b**, The same $4{,}000$ item–replicate pairs partitioned into four termination strata under terminal-aware policy scoring and common replay. Contributions sum to the corresponding all-pair difference. Intervals use a nested item–replicate bootstrap.

a Arm-level estimands

Estimand	Readout context	B	Items	Pairs	Ordinary	Concise/early-answer	Δ [95% CI]
Correct natural completion	Readout-context invariant	512	500	4,000	18.2%	31.9%	+13.8 [+11.6, +16.0]
Correct natural completion	Readout-context invariant	2,048	500	4,000	58.0%	65.6%	+7.6 [+5.8, +9.5]
All-pair accuracy	Terminal-aware policy	512	500	4,000	59.7%	63.5%	+3.8 [+2.1, +5.6]
All-pair accuracy	Terminal-aware policy	2,048	500	4,000	72.1%	73.1%	+1.0 [−0.4, +2.4]
All-pair accuracy	Common replay context	512	500	4,000	59.8%	62.9%	+3.1 [+1.5, +4.8]
All-pair accuracy	Common replay context	2,048	500	4,000	72.0%	73.0%	+1.0 [−0.4, +2.4]
Both-active accuracy	Terminal-aware policy	512	389	2,476	45.2%	49.2%	+4.0 [+1.6, +6.5]
Both-active accuracy	Terminal-aware policy	2,048	132	644	40.2%	41.0%	+0.8 [−4.7, +6.0]
Both-active accuracy	Common replay context	512	389	2,476	45.4%	48.2%	+2.7 [+0.6, +5.0]
Both-active accuracy	Common replay context	2,048	132	644	40.1%	40.5%	+0.5 [−4.8, +5.5]

b Termination-stratum decomposition

Readout context	B	Termination stratum	Share	Within-stratum Δ [95% CI]	Contribution [95% CI]
Terminal-aware policy	512	Neither trace finished	61.9%	+4.0 [+1.6, +6.5]	+2.5 [+1.0, +4.0]
Terminal-aware policy	512	Concise/early-answer finished; ordinary active	17.8%	+8.6 [+4.9, +12.6]	+1.5 [+0.9, +2.3]
Terminal-aware policy	512	Ordinary finished; concise/early-answer active	1.7%	−7.5 [−20.3, +3.8]	−0.1 [−0.4, +0.1]
Terminal-aware policy	512	Both traces finished	18.7%	−0.3 [−1.4, +0.8]	−0.1 [−0.3, +0.2]
Terminal-aware policy	2,048	Neither trace finished	16.1%	+0.8 [−4.7, +6.0]	+0.1 [−0.8, +1.0]
Terminal-aware policy	2,048	Concise/early-answer finished; ordinary active	11.8%	+6.2 [+0.2, +12.1]	+0.7 [+0.0, +1.4]
Terminal-aware policy	2,048	Ordinary finished; concise/early-answer active	1.7%	−16.7 [−34.3, +1.3]	−0.3 [−0.6, +0.0]
Terminal-aware policy	2,048	Both traces finished	70.5%	+0.6 [−0.4, +1.6]	+0.4 [−0.3, +1.2]
Common replay context	512	Neither trace finished	61.9%	+2.7 [+0.6, +5.0]	+1.7 [+0.4, +3.1]
Common replay context	512	Concise/early-answer finished; ordinary active	17.8%	+8.6 [+4.9, +12.6]	+1.5 [+0.9, +2.3]
Common replay context	512	Ordinary finished; concise/early-answer active	1.7%	−6.0 [−19.4, +6.5]	−0.1 [−0.3, +0.1]
Common replay context	512	Both traces finished	18.7%	−0.3 [−1.4, +0.8]	−0.1 [−0.3, +0.2]
Common replay context	2,048	Neither trace finished	16.1%	+0.5 [−4.8, +5.5]	+0.1 [−0.8, +0.9]
Common replay context	2,048	Concise/early-answer finished; ordinary active	11.8%	+6.2 [+0.2, +12.1]	+0.7 [+0.0, +1.4]
Common replay context	2,048	Ordinary finished; concise/early-answer active	1.7%	−16.7 [−34.5, +1.4]	−0.3 [−0.6, +0.0]
Common replay context	2,048	Both traces finished	70.5%	+0.6 [−0.4, +1.7]	+0.5 [−0.3, +1.2]

Table A16: **The 512-token concise/early-answer advantage persists under multiplicity control, direct prompt comparisons, and generated-answer continuation.** **a**, Candidate-logit concise/early-answer minus ordinary generation with simultaneous 95% intervals over the 16 instruction–deadline contrasts in each readout context. **b**, Direct candidate-logit comparisons at $B = 512$. **c**, Greedy generated-answer continuations with $R = 64$ from the stored prefixes. Panels **b** and **c** use pointwise nested item–replicate bootstrap intervals; each cell contains 500 items and eight paired replicates per item.

a Familywise simultaneous inference

Readout context	B	Δ (pp)	Simultaneous 95% CI
Terminal-aware policy	512	+3.8	[+1.7, +6.0]
Terminal-aware policy	2,048	+1.0	[−1.2, +3.1]
Common replay context	512	+3.1	[+1.1, +5.1]
Common replay context	2,048	+1.0	[−1.0, +3.0]

b Direct prompt contrasts at $B = 512$

Readout context	Comparator	Concise/early-answer acc.	Comparator acc.	Δ [95% CI]
Terminal-aware policy	Numeric/concision prompt, $H = 512$	63.5%	60.1%	+3.4 [+1.7, +5.2]
Terminal-aware policy	Numeric/concision prompt, $H = 2{,}048$	63.5%	60.5%	+3.0 [+1.3, +4.8]
Terminal-aware policy	Advance notice of exact stop, $H = 512$	63.5%	61.3%	+2.3 [+0.5, +4.0]
Common replay context	Numeric/concision prompt, $H = 512$	62.9%	60.7%	+2.2 [+0.7, +3.8]
Common replay context	Numeric/concision prompt, $H = 2{,}048$	62.9%	60.6%	+2.4 [+0.9, +3.9]
Common replay context	Advance notice of exact stop, $H = 512$	62.9%	60.6%	+2.3 [+0.8, +3.9]

c Greedy generated-answer continuation

Readout context	B	Concise/early-answer acc.	Ordinary acc.	Δ [95% CI]
Source-policy context	512	63.4%	59.5%	+3.9 [+2.1, +5.7]
Source-policy context	2,048	73.1%	71.0%	+2.1 [+0.6, +3.6]
Common replay context	512	62.8%	59.5%	+3.3 [+1.7, +5.0]
Common replay context	2,048	72.9%	71.0%	+1.9 [+0.4, +3.4]

Table A17: **Generated answers show small estimated benefits from matching the announced number.** Qwen3-14B on 500 MMLU-Pro items with eight paired generations per instruction. The matching contrast averages $H = 512$ minus $H = 2{,}048$ at $B = 512$ and the reverse difference at $B = 2{,}048$. Entries are percentage points. An $R = 64$ answer is generated from every capped reasoning block, including completed blocks, with failures incorrect. Source-policy and common-context continuations are analyzed separately. Pointwise 95% intervals use 10,000 nested item–replicate bootstrap draws preserving the complete instruction–horizon vector. Continuations use token-exact reasoning prefixes. No equivalence margin was specified. The candidate-logit matching estimates are reported in Table A14.

Answer context	Instruction	Matching difference	95% CI
Source-policy context	Numeric/concision	−0.06	[−0.80, +0.68]
Source-policy context	Advance exact-stop notice	+0.20	[−0.50, +0.90]
Common replay context	Numeric/concision	−0.21	[−0.95, +0.54]
Common replay context	Advance exact-stop notice	−0.03	[−0.75, +0.71]

Table A18: **The numeric/concision prompt shortens Qwen reasoning but weakly tracks the announced token count. a**, Budget-specific reasoning length and adherence. Visible denotes the numeric/concision condition and hidden denotes matched ordinary generation. L/B and $P(L \leq B)$ use visible-condition lengths; the log-length contrast is $\log L_{\text{visible}} - \log L_{\text{hidden}}$. **b**, Responsiveness across the four announced budgets. The slope and Spearman correlation are computed within each item–replicate trajectory. Intervals first average the three observed generation replicates within item and then bootstrap items.

a Budget-specific length and adherence

Model	Benchmark	B	Items	Item–replicate pairs	Visible median L	Hidden median L	Median L/B	$P(L \leq B)$	Mean $\Delta \log L$ [95% CI]
Qwen3-14B	GPQA Diamond	512	198	594	3,740.5	4,486.5	7.31	1.9%	-0.179 $[-0.215, -0.144]$
Qwen3-14B	GPQA Diamond	2,048	198	594	3,647.0	4,486.5	1.78	29.6%	-0.176 $[-0.211, -0.142]$
Qwen3-14B	GPQA Diamond	8,192	198	594	3,698.5	4,486.5	0.45	90.6%	-0.159 $[-0.196, -0.121]$
Qwen3-14B	GPQA Diamond	16,384	198	594	3,925.0	4,486.5	0.24	99.2%	-0.133 $[-0.168, -0.099]$
Qwen3-14B	MMLU-Pro	512	500	1,500	884.5	1,094.0	1.73	23.6%	-0.189 $[-0.214, -0.165]$
Qwen3-14B	MMLU-Pro	2,048	500	1,500	880.5	1,094.0	0.43	77.5%	-0.170 $[-0.195, -0.145]$
Qwen3-14B	MMLU-Pro	8,192	500	1,500	929.0	1,094.0	0.11	96.4%	-0.134 $[-0.159, -0.109]$
Qwen3-14B	MMLU-Pro	16,384	500	1,500	905.0	1,094.0	0.06	99.6%	-0.130 $[-0.154, -0.106]$

b Across-budget within-item responsiveness

Model	Benchmark	Metric	Items	Estimate [95% CI]
Qwen3-14B	GPQA Diamond	Within-item slope of $\log L$ on $\log B$	198	$+0.012$ $[+0.004, +0.021]$
Qwen3-14B	GPQA Diamond	Within-item Spearman $\rho(B, L)$	198	$+0.058$ $[+0.013, +0.103]$
Qwen3-14B	MMLU-Pro	Within-item slope of $\log L$ on $\log B$	500	$+0.018$ $[+0.013, +0.024]$
Qwen3-14B	MMLU-Pro	Within-item Spearman $\rho(B, L)$	500	$+0.121$ $[+0.091, +0.150]$

Table A19: **Replicate-specific estimates and crossed-intercept analyses preserve the small, mixed Qwen numeric/concision effects and the positive gpt-oss effort differences.** Pooled rows first average the three observed generation replicates within item and then bootstrap items. The three named replicate rows report separate paired item-bootstrap estimates and 95% intervals. Two-way intervals resample items and replicates. Crossed intervals use method-of-moments item and replicate random intercepts with a Satterthwaite approximation on complete item–replicate matrices, so crossed point estimates can differ from the pooled item-bootstrap Δ when pairs are ineligible. Qwen effects are visible minus hidden; gpt-oss effects are the completed lower-effort readout minus the matched-horizon high-effort readout. All effects are percentage points.

Study	Model	Benchmark	Comparison	Replicate	Items	Item-bootstrap Δ [95% CI]	Two-way item×replicate Δ [95% CI]	Crossed Δ [95% CI]
Qwen textual	Qwen3-14B	GPQA Diamond	Exact, $B = 512$	Pooled (3)	198	-0.5 [-2.5, $+1.3$]	-0.5 [-3.4, $+2.2$]	-0.5 [-2.7, $+1.7$]
				Replicate 1; seed 74,017	198	-1.0 [-5.1, $+3.0$]	—	—
				Replicate 2; seed 74,029	198	$+0.5$ [-3.0, $+4.0$]	—	—
				Replicate 3; seed 74,047	198	-1.0 [-4.5, $+2.5$]	—	—
Qwen textual	Qwen3-14B	GPQA Diamond	Exact, $B = 2{,}048$	Pooled (3)	198	$+0.7$ [-3.0, $+4.4$]	$+0.7$ [-4.0, $+5.7$]	$+0.7$ [-3.3, $+4.7$]
				Replicate 1; seed 74,017	198	-1.5 [-6.6, $+3.5$]	—	—
				Replicate 2; seed 74,029	198	$+0.0$ [-5.1, $+5.1$]	—	—
				Replicate 3; seed 74,047	198	$+3.5$ [-2.0, $+9.6$]	—	—
Qwen textual	Qwen3-14B	GPQA Diamond	Exact, $B = 8{,}192$	Pooled (3)	198	$+0.7$ [-2.4, $+3.9$]	$+0.7$ [-3.4, $+5.1$]	$+0.7$ [-2.5, $+3.8$]
				Replicate 1; seed 74,017	198	$+1.0$ [-4.5, $+6.6$]	—	—
				Replicate 2; seed 74,029	198	-0.5 [-5.1, $+4.0$]	—	—
				Replicate 3; seed 74,047	198	$+1.5$ [-4.0, $+7.1$]	—	—
Qwen textual	Qwen3-14B	GPQA Diamond	Exact, $B = 16{,}384$	Pooled (3)	198	$+1.3$ [-1.9, $+4.5$]	$+1.3$ [-2.9, $+5.7$]	$+1.3$ [-1.8, $+4.5$]
				Replicate 1; seed 74,017	198	$+3.0$ [-2.0, $+8.1$]	—	—
				Replicate 2; seed 74,029	198	$+0.0$ [-5.1, $+5.1$]	—	—
				Replicate 3; seed 74,047	198	$+1.0$ [-4.5, $+6.6$]	—	—
Qwen textual	Qwen3-14B	MMLU-Pro	Exact, $B = 512$	Pooled (3)	500	$+2.0$ [$+0.4$, $+3.7$]	$+2.0$ [-0.1, $+4.1$]	$+2.0$ [$+0.3$, $+3.7$]
				Replicate 1; seed 74,017	500	$+2.6$ [$+0.2$, $+5.2$]	—	—
				Replicate 2; seed 74,029	500	$+1.6$ [-1.0, $+4.2$]	—	—
				Replicate 3; seed 74,047	500	$+1.8$ [-1.0, $+4.6$]	—	—
Qwen textual	Qwen3-14B	MMLU-Pro	Exact, $B = 2{,}048$	Pooled (3)	500	$+0.7$ [-0.9, $+2.1$]	$+0.7$ [-1.6, $+2.9$]	$+0.7$ [-1.1, $+2.5$]
				Replicate 1; seed 74,017	500	$+0.8$ [-1.8, $+3.4$]	—	—
				Replicate 2; seed 74,029	500	$+2.0$ [-0.2, $+4.4$]	—	—
				Replicate 3; seed 74,047	500	-0.8 [-3.0, $+1.6$]	—	—
Qwen textual	Qwen3-14B	MMLU-Pro	Exact, $B = 8{,}192$	Pooled (3)	500	-0.9 [-2.5, $+0.7$]	-0.9 [-3.1, $+1.5$]	-0.9 [-2.5, $+0.7$]
				Replicate 1; seed 74,017	500	-1.0 [-3.6, $+1.6$]	—	—
				Replicate 2; seed 74,029	500	$+0.4$ [-2.2, $+3.0$]	—	—
				Replicate 3; seed 74,047	500	-2.0 [-4.6, $+0.4$]	—	—
Qwen textual	Qwen3-14B	MMLU-Pro	Exact, $B = 16{,}384$	Pooled (3)	500	$+0.1$ [-1.2, $+1.3$]	$+0.1$ [-2.3, $+2.7$]	$+0.1$ [-2.5, $+2.6$]
				Replicate 1; seed 74,017	500	-1.0 [-3.6, $+1.6$]	—	—
				Replicate 2; seed 74,029	500	$+2.2$ [-0.4, $+4.8$]	—	—
				Replicate 3; seed 74,047	500	-1.0 [-3.0, $+1.0$]	—	—

Table A19: **Replicate-specific estimates and crossed-intercept analyses.** Continued: the separate Qwen literal cap-then-answer validation with $R = 16$.

Study	Model	Benchmark	Comparison	Replicate	Items	Item-bootstrap Δ [95% CI]	Two-way item×replicate Δ [95% CI]	Crossed Δ [95% CI]
Qwen literal	Qwen3-14B	GPQA Diamond	Continuation, $B = 512$	Pooled (3)	198	-1.9 [-4.9, $+1.0$]	-1.9 [-5.4, $+1.7$]	-1.9 [-4.8, $+1.1$]
				Replicate 1; seed 74,017	198	-3.0 [-6.6, $+0.5$]	—	—
				Replicate 2; seed 74,029	198	-1.5 [-6.1, $+3.0$]	—	—
				Replicate 3; seed 74,047	198	-1.0 [-5.1, $+3.0$]	—	—
Qwen literal	Qwen3-14B	GPQA Diamond	Continuation, $B = 2{,}048$	Pooled (3)	198	$+0.5$ [-2.7, $+3.7$]	$+0.5$ [-3.9, $+5.4$]	$+0.5$ [-3.4, $+4.5$]
				Replicate 1; seed 74,017	198	-1.5 [-6.1, $+3.0$]	—	—
				Replicate 2; seed 74,029	198	-0.5 [-5.1, $+4.0$]	—	—
				Replicate 3; seed 74,047	198	$+3.5$ [-1.5, $+8.6$]	—	—
Qwen literal	Qwen3-14B	MMLU-Pro	Continuation, $B = 512$	Pooled (3)	500	$+2.0$ [$+0.2$, $+3.9$]	$+2.0$ [-0.4, $+4.5$]	$+2.0$ [$+0.2$, $+3.8$]
				Replicate 1; seed 74,017	500	$+2.0$ [-0.6, $+4.6$]	—	—
				Replicate 2; seed 74,029	500	$+0.8$ [-2.0, $+3.6$]	—	—
				Replicate 3; seed 74,047	500	$+3.2$ [$+0.2$, $+6.2$]	—	—
Qwen literal	Qwen3-14B	MMLU-Pro	Continuation, $B = 2{,}048$	Pooled (3)	500	$+0.1$ [-1.3, $+1.5$]	$+0.1$ [-2.3, $+2.4$]	$+0.1$ [-2.1, $+2.3$]
				Replicate 1; seed 74,017	500	-1.0 [-3.4, $+1.6$]	—	—
				Replicate 2; seed 74,029	500	-0.8 [-3.2, $+1.6$]	—	—
				Replicate 3; seed 74,047	500	$+2.0$ [-0.2, $+4.4$]	—	—

Table A19: **Replicate-specific estimates and crossed-intercept analyses.** Continued: gpt-oss candidate-logit comparisons.

Study	Model	Benchmark	Comparison	Replicate	Items	Item-bootstrap Δ [95% CI]	Two-way item×replicate Δ [95% CI]	Crossed Δ [95% CI]
gpt-oss exact	gpt-oss-20b	GPQA Diamond	low	Pooled (3)	198	+14.5 [+7.7, +21.2]	+14.5 [+6.1, +22.7]	+14.5 [+6.9, +22.0]
				Replicate 1; seed 74,017	198	+13.6 [+5.1, +22.2]	—	—
				Replicate 2; seed 74,029	198	+10.6 [+1.5, +19.7]	—	—
				Replicate 3; seed 74,047	198	+19.2 [+10.6, +27.8]	—	—
gpt-oss exact	gpt-oss-20b	GPQA Diamond	medium	Pooled (3)	198	+17.4 [+11.3, +23.7]	+17.4 [+9.6, +25.8]	+17.8 [+11.1, +24.4]
				Replicate 1; seed 74,017	197	+13.7 [+5.6, +22.3]	—	—
				Replicate 2; seed 74,029	198	+17.7 [+9.6, +25.8]	—	—
				Replicate 3; seed 74,047	198	+21.2 [+12.6, +29.8]	—	—
gpt-oss exact	gpt-oss-20b	MMLU-Pro	low	Pooled (3)	500	+20.9 [+16.7, +25.0]	+20.9 [+16.2, +25.5]	+20.9 [+16.8, +25.1]
				Replicate 1; seed 74,017	500	+19.4 [+14.2, +24.4]	—	—
				Replicate 2; seed 74,029	500	+21.2 [+16.2, +26.0]	—	—
				Replicate 3; seed 74,047	500	+22.2 [+17.2, +27.0]	—	—
gpt-oss exact	gpt-oss-20b	MMLU-Pro	medium	Pooled (3)	500	+18.0 [+14.5, +21.5]	+18.0 [+13.7, +22.3]	+18.0 [+14.5, +21.5]
				Replicate 1; seed 74,017	500	+17.2 [+12.4, +21.8]	—	—
				Replicate 2; seed 74,029	500	+17.0 [+12.2, +21.8]	—	—
				Replicate 3; seed 74,047	500	+19.8 [+14.8, +24.6]	—	—
gpt-oss exact	gpt-oss-120b	GPQA Diamond	low	Pooled (3)	198	+23.6 [+16.3, +30.6]	+23.6 [+15.7, +31.5]	+23.6 [+16.3, +30.9]
				Replicate 1; seed 74,017	198	+20.7 [+12.1, +29.3]	—	—
				Replicate 2; seed 74,029	198	+26.3 [+18.2, +34.3]	—	—
				Replicate 3; seed 74,047	198	+23.7 [+15.7, +31.8]	—	—
gpt-oss exact	gpt-oss-120b	GPQA Diamond	medium	Pooled (3)	198	+23.6 [+16.7, +30.5]	+23.6 [+16.2, +31.0]	+23.6 [+16.6, +30.5]
				Replicate 1; seed 74,017	198	+23.7 [+15.6, +31.8]	—	—
				Replicate 2; seed 74,029	198	+24.2 [+16.2, +32.3]	—	—
				Replicate 3; seed 74,047	198	+22.7 [+14.6, +30.8]	—	—
gpt-oss exact	gpt-oss-120b	MMLU-Pro	low	Pooled (3)	500	+26.3 [+21.9, +30.5]	+26.3 [+21.7, +30.7]	+26.3 [+22.0, +30.6]
				Replicate 1; seed 74,017	500	+27.4 [+22.8, +32.0]	—	—
				Replicate 2; seed 74,029	500	+25.4 [+20.6, +30.0]	—	—
				Replicate 3; seed 74,047	500	+26.0 [+21.0, +30.8]	—	—
gpt-oss exact	gpt-oss-120b	MMLU-Pro	medium	Pooled (3)	500	+24.2 [+20.3, +28.2]	+24.2 [+19.6, +28.9]	+24.2 [+19.9, +28.5]
				Replicate 1; seed 74,017	500	+21.8 [+17.4, +26.2]	—	—
				Replicate 2; seed 74,029	500	+25.8 [+21.2, +30.6]	—	—
				Replicate 3; seed 74,047	500	+25.0 [+20.2, +29.8]	—	—

Table A19: **Replicate-specific estimates and crossed-intercept analyses.** Continued: gpt-oss generated-answer continuations.

Study	Model	Benchmark	Comparison	Replicate	Items	Item-bootstrap Δ [95% CI]	Two-way item×replicate Δ [95% CI]	Crossed Δ [95% CI]
gpt-oss continuation	gpt-oss-20b	GPQA Diamond	low	Pooled (3)	198	+15.5 [+8.8, +22.1]	+15.5 [+7.6, +23.2]	+15.5 [+8.8, +22.2]
				Replicate 1; seed 74,017	198	+16.7 [+8.6, +24.7]	—	—
				Replicate 2; seed 74,029	198	+12.1 [+3.5, +20.7]	—	—
				Replicate 3; seed 74,047	198	+17.7 [+9.1, +26.3]	—	—
gpt-oss continuation	gpt-oss-20b	GPQA Diamond	medium	Pooled (3)	198	+17.4 [+11.1, +23.8]	+17.4 [+9.8, +25.3]	+17.8 [+11.4, +24.1]
				Replicate 1; seed 74,017	197	+14.7 [+6.1, +23.4]	—	—
				Replicate 2; seed 74,029	198	+17.2 [+9.1, +25.3]	—	—
				Replicate 3; seed 74,047	198	+20.7 [+12.1, +29.3]	—	—
gpt-oss continuation	gpt-oss-20b	MMLU-Pro	low	Pooled (3)	500	+20.5 [+16.2, +24.5]	+20.5 [+15.9, +24.9]	+20.5 [+16.3, +24.7]
				Replicate 1; seed 74,017	500	+19.4 [+14.2, +24.4]	—	—
				Replicate 2; seed 74,029	500	+20.6 [+15.6, +25.6]	—	—
				Replicate 3; seed 74,047	500	+21.4 [+16.4, +26.0]	—	—
gpt-oss continuation	gpt-oss-20b	MMLU-Pro	medium	Pooled (3)	500	+18.2 [+14.7, +21.7]	+18.2 [+13.9, +22.5]	+18.2 [+14.8, +21.6]
				Replicate 1; seed 74,017	500	+17.2 [+12.4, +21.8]	—	—
				Replicate 2; seed 74,029	500	+17.4 [+12.6, +22.0]	—	—
				Replicate 3; seed 74,047	500	+20.0 [+15.0, +24.8]	—	—
gpt-oss continuation	gpt-oss-120b	GPQA Diamond	low	Pooled (3)	198	+23.1 [+15.8, +30.3]	+23.1 [+15.2, +30.8]	+23.1 [+15.8, +30.3]
				Replicate 1; seed 74,017	198	+20.2 [+11.6, +28.8]	—	—
				Replicate 2; seed 74,029	198	+24.2 [+16.2, +32.3]	—	—
				Replicate 3; seed 74,047	198	+24.7 [+16.7, +32.8]	—	—
gpt-oss continuation	gpt-oss-120b	GPQA Diamond	medium	Pooled (3)	198	+23.1 [+16.2, +30.0]	+23.1 [+15.8, +30.3]	+23.1 [+16.2, +29.9]
				Replicate 1; seed 74,017	198	+23.2 [+15.2, +31.3]	—	—
				Replicate 2; seed 74,029	198	+23.2 [+15.2, +31.3]	—	—
				Replicate 3; seed 74,047	198	+22.7 [+14.6, +30.8]	—	—
gpt-oss continuation	gpt-oss-120b	MMLU-Pro	low	Pooled (3)	500	+25.9 [+21.5, +30.2]	+25.9 [+21.3, +30.4]	+25.9 [+21.6, +30.3]
				Replicate 1; seed 74,017	500	+26.2 [+21.4, +31.0]	—	—
				Replicate 2; seed 74,029	500	+26.4 [+21.4, +31.2]	—	—
				Replicate 3; seed 74,047	500	+25.2 [+20.4, +30.0]	—	—
gpt-oss continuation	gpt-oss-120b	MMLU-Pro	medium	Pooled (3)	500	+24.1 [+20.1, +28.0]	+24.1 [+19.5, +28.7]	+24.1 [+19.9, +28.3]
				Replicate 1; seed 74,017	500	+22.0 [+17.6, +26.4]	—	—
				Replicate 2; seed 74,029	500	+26.0 [+21.4, +30.6]	—	—
				Replicate 3; seed 74,047	500	+24.2 [+19.4, +29.0]	—	—

Table A20: **Selected answer-context checks do not reverse the main interpretation.** Qwen neutral replay applies the same budget-free closure to every prefix, including completed traces. The second block holds the source prefix fixed and changes only the answer-time context. All rows use full-vocabulary candidate logits and paired item-bootstrap intervals. Limited-time, paraphrased, and neutral Qwen closure wording gives the same qualitative pattern (Table A28).

Family	Contrast	Benchmark	B	n	Δ (pp)	95% CI
Neutral replay contrasts						
Qwen3-14B	prompt vs. surprise, neutral replay	GPQA Diamond	512	198	-1.3	$[-5.4, +2.5]$
Qwen3-14B	prompt vs. surprise, neutral replay	GPQA Diamond	2,048	198	-0.3	$[-4.4, +3.7]$
Qwen3-14B	prompt vs. surprise, neutral replay	GPQA Diamond	8,192	198	-3.0	$[-6.1, +0.0]$
Qwen3-14B	prompt vs. surprise, neutral replay	GPQA Diamond	16,384	198	$+1.9$	$[-1.6, +5.6]$
Qwen3-14B	prompt vs. surprise, neutral replay	MMLU-Pro	512	500	$+1.9$	$[+0.1, +3.7]$
Qwen3-14B	prompt vs. surprise, neutral replay	MMLU-Pro	2,048	500	$+1.1$	$[-0.5, +2.7]$
Qwen3-14B	prompt vs. surprise, neutral replay	MMLU-Pro	8,192	500	$+1.0$	$[-0.6, +2.7]$
Qwen3-14B	prompt vs. surprise, neutral replay	MMLU-Pro	16,384	500	$+0.8$	$[-0.9, +2.5]$
Direct effects of the readout context						
Qwen3-14B	budget present vs. absent at readout	GPQA Diamond	512	198	-2.9	$[-5.9, +0.0]$
Qwen3-14B	budget present vs. absent at readout	GPQA Diamond	2,048	198	-0.5	$[-2.9, +1.9]$
Qwen3-14B	budget present vs. absent at readout	MMLU-Pro	512	500	-0.4	$[-1.5, +0.7]$
Qwen3-14B	budget present vs. absent at readout	MMLU-Pro	2,048	500	$+0.1$	$[-0.2, +0.3]$
gpt-oss-20b	low vs. medium readout, high source	GPQA Diamond	512	191	$+2.1$	$[-1.6, +6.3]$
gpt-oss-20b	low vs. medium readout, high source	MMLU-Pro	512	361	$+1.7$	$[+0.0, +3.6]$
gpt-oss-120b	low vs. medium readout, high source	GPQA Diamond	512	188	$+0.0$	$[-2.7, +2.7]$
gpt-oss-120b	low vs. medium readout, high source	MMLU-Pro	512	335	$+0.3$	$[-2.1, +2.7]$

Table A21: **Qwen proper scores show no consistent advantage for the numeric/concision prompt at 512 or 2,048 tokens.** Visible denotes the numeric/concision condition and hidden the matched hidden-horizon condition. Δ is visible minus hidden. Higher valid-option mass is preferred; lower conditional log loss and Brier score are preferred. Rows include token-exact candidate-logit pairs from three generation replicates. Arm means and differences first average replicates within item, after which intervals bootstrap items.

Model	Benchmark	B	Metric	Items	Item–replicate pairs	Replicates	Visible	Hidden	Δ [95% CI]
Qwen3-14B	GPQA Diamond	512	Valid-option mass	197	580	3	99.6%	99.5%	+0.1 [−0.2, +0.3]
Qwen3-14B	GPQA Diamond	512	Conditional log loss	197	580	3	3.310	3.021	+0.289 [+0.142, +0.459]
Qwen3-14B	GPQA Diamond	512	Conditional Brier score	197	580	3	0.942	0.918	+0.024 [+0.001, +0.048]
Qwen3-14B	GPQA Diamond	2,048	Valid-option mass	146	394	3	99.1%	99.1%	+0.1 [−0.3, +0.4]
Qwen3-14B	GPQA Diamond	2,048	Conditional log loss	146	394	3	2.869	2.738	+0.131 [−0.176, +0.470]
Qwen3-14B	GPQA Diamond	2,048	Conditional Brier score	146	394	3	0.918	0.923	−0.005 [−0.076, +0.070]
Qwen3-14B	MMLU-Pro	512	Valid-option mass	406	1,085	3	96.9%	96.5%	+0.4 [−0.1, +0.9]
Qwen3-14B	MMLU-Pro	512	Conditional log loss	406	1,085	3	4.234	4.351	−0.117 [−0.324, +0.075]
Qwen3-14B	MMLU-Pro	512	Conditional Brier score	406	1,085	3	0.822	0.861	−0.039 [−0.078, −0.003]
Qwen3-14B	MMLU-Pro	2,048	Valid-option mass	132	299	3	93.9%	93.8%	+0.1 [−1.5, +1.7]
Qwen3-14B	MMLU-Pro	2,048	Conditional log loss	132	299	3	3.645	3.811	−0.167 [−0.666, +0.341]
Qwen3-14B	MMLU-Pro	2,048	Conditional Brier score	132	299	3	0.877	0.898	−0.021 [−0.112, +0.070]

Table A22: **Lower gpt-oss effort is more accurate at its stopping point under candidate-logit and generated-answer readouts.** Each row compares a completed low- or medium-effort reasoning block with high effort at the item- and replicate-specific lower-effort terminal horizon: 198 GPQA Diamond or 500 MMLU-Pro items across three generation replicates. If high effort finishes before that horizon, its completed reasoning block is replayed. Candidate-logit and high-active columns use a common medium-effort answer context; the latter restricts to pairs in which high effort remains unfinished. Generated-answer columns continue all replayable prefixes without option normalization. Active/all gives the high-active and replayable pair counts, and parse coverage is the share with two parsed continuations. Differences are the completed lower-effort readout minus the matched-horizon high-effort readout. Intervals first average replicates within item and then bootstrap items (10,000 resamples).

Model	Benchmark	Effort	Candidate-logit Δ [95% CI]	High-active Δ [95% CI]	Active / all pairs	Generated-answer Δ [95% CI]	Parse coverage
gpt-oss-20b	GPQA Diamond	low	+14.5 [+7.7, +21.2]	+14.5 [+7.7, +21.2]	594/594	+15.5 [+8.8, +22.1]	99.7%
gpt-oss-20b	GPQA Diamond	medium	+17.4 [+11.3, +23.7]	+17.9 [+11.5, +24.3]	567/593	+17.4 [+11.1, +23.8]	98.8%
gpt-oss-20b	MMLU-Pro	low	+20.9 [+16.7, +25.0]	+21.1 [+16.9, +25.2]	1,496/1,500	+20.5 [+16.2, +24.5]	99.3%
gpt-oss-20b	MMLU-Pro	medium	+18.0 [+14.5, +21.5]	+19.2 [+15.6, +22.9]	1,407/1,500	+18.2 [+14.7, +21.7]	99.1%
gpt-oss-120b	GPQA Diamond	low	+23.6 [+16.3, +30.6]	+23.6 [+16.3, +30.6]	594/594	+23.1 [+15.8, +30.3]	99.8%
gpt-oss-120b	GPQA Diamond	medium	+23.6 [+16.7, +30.5]	+23.9 [+17.0, +30.9]	591/594	+23.1 [+16.2, +30.0]	100.0%
gpt-oss-120b	MMLU-Pro	low	+26.3 [+21.9, +30.5]	+26.3 [+21.9, +30.5]	1,499/1,500	+25.9 [+21.5, +30.2]	99.7%
gpt-oss-120b	MMLU-Pro	medium	+24.2 [+20.3, +28.2]	+24.9 [+20.9, +28.9]	1,473/1,500	+24.1 [+20.1, +28.0]	99.5%

Table A23: **High valid-option mass coexists with proper-score disagreement driven by near-zero correct-option probabilities in terminal readouts.** The table uses the all-replayable token-exact prefixes in Table A22, comprising 198 GPQA Diamond or 500 MMLU-Pro items across three generation replicates. **a**, Valid-option mass. **b**, Absolute arm means and paired differences for conditional log loss and multiclass Brier score. **c**, Median conditional log loss and the share of all probes with option-normalized $q_{\text{gold}} < 10^{-4}$. Differences are the completed lower-effort readout minus the matched-horizon high-effort readout. Higher valid-option mass and lower proper scores are preferred. Means and differences first average replicates within item; intervals then bootstrap items.

a Valid-option mass

Model	Benchmark	Effort	Items	Replicates	Terminal	High prefix	Δ [95% CI]
gpt-oss-20b	GPQA Diamond	low	198	3	99.9%	98.5%	+1.5 [+1.2, +1.8]
gpt-oss-20b	GPQA Diamond	medium	198	3	99.8%	98.0%	+1.8 [+1.3, +2.5]
gpt-oss-20b	MMLU-Pro	low	500	3	99.6%	99.2%	+0.4 [−0.0, +0.8]
gpt-oss-20b	MMLU-Pro	medium	500	3	99.9%	99.3%	+0.6 [+0.4, +0.8]
gpt-oss-120b	GPQA Diamond	low	198	3	100.0%	100.0%	+0.0 [+0.0, +0.0]
gpt-oss-120b	GPQA Diamond	medium	198	3	100.0%	100.0%	+0.0 [+0.0, +0.0]
gpt-oss-120b	MMLU-Pro	low	500	3	100.0%	99.9%	+0.1 [+0.0, +0.2]
gpt-oss-120b	MMLU-Pro	medium	500	3	100.0%	99.9%	+0.1 [+0.0, +0.2]

b Conditional proper-score means

Model	Benchmark	Effort	Metric	Terminal	High prefix	Δ [95% CI]
gpt-oss-20b	GPQA Diamond	low	Conditional log loss	5.732	1.396	+4.336 [+3.599, +5.068]
			Conditional Brier score	0.878	0.723	+0.155 [+0.051, +0.259]
gpt-oss-20b	GPQA Diamond	medium	Conditional log loss	5.803	1.748	+4.056 [+3.224, +4.890]
			Conditional Brier score	0.702	0.725	−0.023 [−0.124, +0.079]
gpt-oss-20b	MMLU-Pro	low	Conditional log loss	5.314	1.855	+3.459 [+2.953, +3.987]
			Conditional Brier score	0.712	0.721	−0.009 [−0.072, +0.056]
gpt-oss-20b	MMLU-Pro	medium	Conditional log loss	5.075	2.328	+2.747 [+2.241, +3.264]
			Conditional Brier score	0.552	0.651	−0.099 [−0.154, −0.045]
gpt-oss-120b	GPQA Diamond	low	Conditional log loss	8.870	3.351	+5.520 [+4.126, +6.923]
			Conditional Brier score	0.700	1.014	−0.314 [−0.449, −0.179]
gpt-oss-120b	GPQA Diamond	medium	Conditional log loss	8.457	3.286	+5.171 [+3.687, +6.648]
			Conditional Brier score	0.630	0.957	−0.327 [−0.460, −0.196]
gpt-oss-120b	MMLU-Pro	low	Conditional log loss	6.219	2.974	+3.244 [+2.429, +4.068]
			Conditional Brier score	0.491	0.812	−0.322 [−0.398, −0.245]
gpt-oss-120b	MMLU-Pro	medium	Conditional log loss	5.623	3.196	+2.427 [+1.655, +3.226]
			Conditional Brier score	0.419	0.759	−0.341 [−0.413, −0.270]

c Conditional log-loss medians and lower tails

Model	Benchmark	Effort	Pairs	Terminal median	High-prefix median	$q_{\text{gold}} < 10^{-4}$, terminal / high prefix
gpt-oss-20b	GPQA Diamond	low	594	0.000	1.224	36.0% / 0.2%
gpt-oss-20b	GPQA Diamond	medium	593	0.000	1.155	34.6% / 2.4%
gpt-oss-20b	MMLU-Pro	low	1,500	0.000	1.634	32.3% / 0.4%
gpt-oss-20b	MMLU-Pro	medium	1,500	0.000	0.837	27.4% / 5.9%
gpt-oss-120b	GPQA Diamond	low	594	0.000	2.144	34.8% / 8.2%
gpt-oss-120b	GPQA Diamond	medium	594	0.000	1.542	31.5% / 8.6%
gpt-oss-120b	MMLU-Pro	low	1,500	0.000	0.946	24.5% / 8.5%
gpt-oss-120b	MMLU-Pro	medium	1,500	0.000	0.259	20.9% / 11.7%

Table A24: **The fixed-horizon gpt-oss advantage is concentrated where lower effort has completed and high effort remains active.** Active means that reasoning had not terminated at $B = 512$. Unfinished prefixes use common medium-effort candidate-logit scoring, and completed runs retain their natural answer. Shares and contributions use item–replicate pairs from three generated replicates, and contributions sum to the overall lower-minus-high accuracy difference. Within-stratum intervals average replicate outcomes within item before bootstrapping items. Because contributions use raw pair means, they need not equal the share times the displayed item-averaged difference.

Model	Benchmark	Contrast	Termination stratum	Items	Pairs	Share	Within-stratum Δ [95% CI]	Contribution
gpt-oss-20b	GPQA Diamond	Low–high	Both active	95	178	30.0%	−4.2 [−13.3, +4.9]	−1.0
gpt-oss-20b	GPQA Diamond	Low–high	Lower complete; high active	171	403	67.8%	+16.2 [+8.1, +24.2]	+11.8
gpt-oss-20b	GPQA Diamond	Low–high	Both complete	7	13	2.2%	−14.3 [−42.9, +0.0]	−0.2
gpt-oss-20b	GPQA Diamond	Medium–high	Both active	182	501	84.3%	+2.8 [−1.5, +7.1]	+1.9
gpt-oss-20b	GPQA Diamond	Medium–high	Lower complete; high active	49	80	13.5%	+23.1 [+12.2, +34.7]	+3.4
gpt-oss-20b	GPQA Diamond	Medium–high	Both complete	7	13	2.2%	+0.0 [+0.0, +0.0]	+0.0
gpt-oss-20b	MMLU-Pro	Low–high	Both active	56	92	6.2%	+1.2 [−11.0, +13.4]	−0.1
gpt-oss-20b	MMLU-Pro	Low–high	Lower complete; high active	416	1,045	69.9%	+7.3 [+3.3, +11.3]	+6.2
gpt-oss-20b	MMLU-Pro	Low–high	Both complete	169	357	23.9%	−6.8 [−10.5, −3.5]	−1.5
gpt-oss-20b	MMLU-Pro	Medium–high	Both active	303	710	47.4%	−1.6 [−5.7, +2.6]	−0.3
gpt-oss-20b	MMLU-Pro	Medium–high	Lower complete; high active	255	432	28.8%	+11.6 [+6.9, +16.3]	+3.4
gpt-oss-20b	MMLU-Pro	Medium–high	Both complete	163	344	22.9%	−0.4 [−2.5, +1.6]	−0.1
gpt-oss-20b	MMLU-Pro	Medium–high	High complete; lower active	12	13	0.9%	−25.0 [−50.0, +0.0]	−0.2
gpt-oss-120b	GPQA Diamond	Low–high	Both active	63	132	22.2%	+10.6 [−1.6, +22.8]	+3.2
gpt-oss-120b	GPQA Diamond	Low–high	Lower complete; high active	168	438	73.7%	+22.3 [+15.2, +29.5]	+15.8
gpt-oss-120b	GPQA Diamond	Low–high	Both complete	13	24	4.0%	+0.0 [+0.0, +0.0]	+0.0
gpt-oss-120b	GPQA Diamond	Medium–high	Both active	154	421	70.9%	+4.5 [−0.1, +9.4]	+1.9
gpt-oss-120b	GPQA Diamond	Medium–high	Lower complete; high active	68	149	25.1%	+20.1 [+9.8, +30.4]	+5.6
gpt-oss-120b	GPQA Diamond	Medium–high	Both complete	13	24	4.0%	+0.0 [+0.0, +0.0]	+0.0
gpt-oss-120b	MMLU-Pro	Low–high	Both active	46	99	6.6%	+6.5 [−6.5, +19.6]	+0.4
gpt-oss-120b	MMLU-Pro	Low–high	Lower complete; high active	353	900	60.0%	+12.0 [+7.7, +16.3]	+7.4
gpt-oss-120b	MMLU-Pro	Low–high	Both complete	214	500	33.4%	+0.2 [−2.1, +2.6]	−0.1
gpt-oss-120b	MMLU-Pro	Medium–high	Both active	164	379	25.3%	+2.0 [−3.5, +7.5]	+0.0
gpt-oss-120b	MMLU-Pro	Medium–high	Lower complete; high active	284	621	41.4%	+11.4 [+7.0, +15.9]	+5.3
gpt-oss-120b	MMLU-Pro	Medium–high	Both complete	213	498	33.2%	+0.9 [−0.9, +2.8]	+0.1
gpt-oss-120b	MMLU-Pro	Medium–high	High complete; lower active	2	2	0.1%	−50.0 [−100.0, +0.0]	−0.1

Table A25: **Jointly unfinished gpt-oss contrasts under item and pair weighting.** Lower-minus-high candidate-logit accuracy differences at $B = 512$ among pairs in which neither run had finished, from the same three-replicate cohorts as Table A24. The item-weighted contrast averages replicate outcomes within item and then across items, as displayed in Table A24. The pair-weighted contrast is the raw mean over pairs, so its product with the stratum share equals the stratum's contribution to the overall difference. Intervals are pointwise 95% item-bootstrap intervals from $10{,}000$ draws that resample items with all their pairs.

Model	Benchmark	Comparison	Pairs	Items	Share	Item-weighted Δ [95% CI]	Pair-weighted Δ [95% CI]	Contribution
gpt-oss-20b	GPQA Diamond	Low minus high	178	95	30.0%	−4.2 [−13.3, +4.9]	−3.4 [−11.9, +5.2]	−1.0
gpt-oss-20b	GPQA Diamond	Medium minus high	501	182	84.3%	+2.8 [−1.5, +7.1]	+2.2 [−2.0, +6.3]	+1.9
gpt-oss-20b	MMLU-Pro	Low minus high	92	56	6.2%	+1.2 [−11.0, +13.4]	−1.1 [−11.0, +8.5]	−0.1
gpt-oss-20b	MMLU-Pro	Medium minus high	710	303	47.4%	−1.6 [−5.7, +2.6]	−0.7 [−4.4, +3.0]	−0.3
gpt-oss-120b	GPQA Diamond	Low minus high	132	63	22.2%	+10.6 [−1.6, +22.8]	+14.4 [+3.8, +25.2]	+3.2
gpt-oss-120b	GPQA Diamond	Medium minus high	421	154	70.9%	+4.5 [−0.1, +9.4]	+2.6 [−1.4, +6.8]	+1.9
gpt-oss-120b	MMLU-Pro	Low minus high	99	46	6.6%	+6.5 [−6.5, +19.6]	+6.1 [−6.7, +19.0]	+0.4
gpt-oss-120b	MMLU-Pro	Medium minus high	379	164	25.3%	+2.0 [−3.5, +7.5]	+0.0 [−5.2, +5.3]	+0.0

Table A26: **Lower gpt-oss effort makes answer evidence available earlier, while active-prefix differences are smaller.** The terminal-aware area under the curve (AUC) averages paired correct-option log-odds differences from 0 to 512 reasoning tokens and carries terminal states forward. Stable-correct Δ_{256} compares whether answers at 256 tokens remain correct through 512. The active-pair AUC retains item–replicate pairs for which both traces have $L > 128$ and recomputes the AUC over 0–128; conditioning on survival changes the compared population. Qwen contrasts are numeric/concision prompt minus surprise stop. gpt-oss contrasts are lower effort minus high effort. Intervals bootstrap items.

a Terminal-aware and stable-correct contrasts

Model	Benchmark	Contrast	n	Terminal-aware AUC Δ log-odds	Stable-correct Δ_{256} (pp)
gpt-oss-20b	GPQA Diamond	low minus high	196	$+1.85$ $[+0.74, +2.90]$	$+15.8$ $[+8.2, +23.5]$
gpt-oss-20b	GPQA Diamond	medium minus high	196	$+0.83$ $[+0.49, +1.18]$	$+4.3$ $[+0.2, +8.3]$
gpt-oss-20b	MMLU-Pro	low minus high	500	$+1.44$ $[+0.76, +2.11]$	$+17.7$ $[+14.1, +21.3]$
gpt-oss-20b	MMLU-Pro	medium minus high	500	$+1.36$ $[+1.04, +1.70]$	$+8.3$ $[+5.8, +10.7]$
gpt-oss-120b	GPQA Diamond	low minus high	198	$+4.22$ $[+2.38, +6.00]$	$+14.1$ $[+7.6, +20.7]$
gpt-oss-120b	GPQA Diamond	medium minus high	198	$+2.00$ $[+1.29, +2.70]$	$+3.9$ $[-0.5, +8.4]$
gpt-oss-120b	MMLU-Pro	low minus high	500	$+3.76$ $[+2.59, +4.91]$	$+15.7$ $[+11.7, +19.6]$
gpt-oss-120b	MMLU-Pro	medium minus high	500	$+3.27$ $[+2.65, +3.87]$	$+9.2$ $[+6.5, +11.9]$
Qwen3-14B	GPQA Diamond	prompt minus surprise	198	$+0.17$ $[+0.03, +0.32]$	-1.2 $[-4.2, +1.9]$
Qwen3-14B	MMLU-Pro	prompt minus surprise	500	$+0.42$ $[+0.18, +0.65]$	$+0.8$ $[-0.7, +2.3]$

b Jointly active early-prefix contrasts

Model	Benchmark	Contrast	Active item–replicate pairs	Active-pair $AUC_{0:128}$
gpt-oss-20b	GPQA Diamond	low minus high	414	$+0.05$ $[-0.10, +0.20]$
gpt-oss-20b	GPQA Diamond	medium minus high	580	$+0.06$ $[-0.02, +0.16]$
gpt-oss-20b	MMLU-Pro	low minus high	498	$+0.12$ $[-0.06, +0.28]$
gpt-oss-20b	MMLU-Pro	medium minus high	1,314	$+0.19$ $[+0.04, +0.36]$
gpt-oss-120b	GPQA Diamond	low minus high	441	-0.10 $[-0.46, +0.25]$
gpt-oss-120b	GPQA Diamond	medium minus high	580	-0.05 $[-0.24, +0.15]$
gpt-oss-120b	MMLU-Pro	low minus high	534	$+0.12$ $[-0.33, +0.55]$
gpt-oss-120b	MMLU-Pro	medium minus high	1,178	$+0.53$ $[+0.29, +0.77]$
Qwen3-14B	GPQA Diamond	prompt minus surprise	593	$+0.08$ $[-0.01, +0.18]$
Qwen3-14B	MMLU-Pro	prompt minus surprise	1,498	$+0.06$ $[-0.03, +0.16]$

Table A27: **Qwen's small prompt effects are distributed across unfinished-prefix and completion strata.** Qwen3-14B numeric/concision-prompt and hidden-horizon traces are partitioned by whether each arm has completed at the imposed horizon B. Shares use all token-exact matched item–replicate pairs and sum to one before rounding within each benchmark–horizon cell. Within-stratum differences are visible minus hidden accuracy. Contributions use the item-normalized pair decomposition and sum before rounding to the pooled terminal-aware candidate-logit difference. Intervals average the three observed replicates within item before bootstrapping items. All four horizons are reported. Several late-horizon strata, especially the both-active strata, are sparse and should be read descriptively.

Benchmark	B	Termination stratum	Items	Item–replicate pairs	Share	Within-stratum Δ [95% CI]	Contribution
GPQA Diamond	512	Both active	197	580	97.6%	-0.8 $[-2.8, +1.2]$	-0.7
		Visible complete; hidden active	9	11	1.9%	$+11.1$ $[+0.0, +33.3]$	$+0.2$
		Hidden complete; visible active	3	3	0.5%	$+0.0$ $[+0.0, +0.0]$	$+0.0$
GPQA Diamond	2,048	Both active	146	394	66.3%	-0.5 $[-5.8, +4.9]$	-0.3
		Visible complete; hidden active	42	56	9.4%	$+8.3$ $[+2.4, +15.5]$	$+0.8$
		Hidden complete; visible active	22	24	4.0%	$+2.3$ $[-13.6, +18.2]$	$+0.0$
		Both complete	55	120	20.2%	$+1.2$ $[-1.8, +5.5]$	$+0.2$
GPQA Diamond	8,192	Both active	25	41	6.9%	$+7.3$ $[-9.3, +24.0]$	$+0.7$
		Visible complete; hidden active	42	56	9.4%	$+0.4$ $[-15.1, +15.1]$	-0.2
		Hidden complete; visible active	12	15	2.5%	-33.3 $[-58.3, -8.3]$	-0.8
		Both complete	180	482	81.1%	$+1.0$ $[-2.2, +4.3]$	$+1.0$
GPQA Diamond	16,384	Both active	1	2	0.3%	$+0.0$ $[+0.0, +0.0]$	$+0.0$
		Visible complete; hidden active	2	3	0.5%	$+50.0$ $[+0.0, +100.0]$	$+0.3$
		Hidden complete; visible active	3	3	0.5%	$+0.0$ $[-100.0, +100.0]$	$+0.0$
		Both complete	197	586	98.7%	$+0.7$ $[-2.5, +3.9]$	$+1.0$
MMLU-Pro	512	Both active	406	1,085	72.3%	$+2.0$ $[-0.1, +4.2]$	$+1.3$
		Visible complete; hidden active	99	118	7.9%	$+6.6$ $[+0.5, +13.1]$	$+0.5$
		Hidden complete; visible active	54	61	4.1%	$+0.9$ $[-6.5, +8.3]$	$+0.1$
		Both complete	118	236	15.7%	$+0.6$ $[-2.0, +3.1]$	$+0.1$
MMLU-Pro	2,048	Both active	132	299	19.9%	$+2.0$ $[-3.5, +7.6]$	$+0.2$
		Visible complete; hidden active	102	146	9.7%	$+2.6$ $[-4.7, +9.8]$	$+0.3$
		Hidden complete; visible active	35	39	2.6%	-5.7 $[-18.6, +7.1]$	-0.2
		Both complete	376	1,016	67.7%	$+1.3$ $[-0.1, +2.9]$	$+0.4$
MMLU-Pro	8,192	Both active	19	40	2.7%	-11.4 $[-28.1, +2.6]$	-0.3
		Visible complete; hidden active	29	34	2.3%	$+1.7$ $[-17.2, +20.7]$	$+0.1$
		Hidden complete; visible active	11	14	0.9%	-22.7 $[-54.5, +4.5]$	-0.2
		Both complete	481	1,412	94.1%	-0.8 $[-2.5, +0.8]$	-0.5
MMLU-Pro	16,384	Both active	2	3	0.2%	-50.0 $[-100.0, +0.0]$	-0.1
		Visible complete; hidden active	5	7	0.5%	$+20.0$ $[+0.0, +40.0]$	$+0.1$
		Hidden complete; visible active	2	3	0.2%	-50.0 $[-100.0, +0.0]$	-0.1
		Both complete	498	1,487	99.1%	$+0.1$ $[-1.3, +1.5]$	$+0.1$

Table A28: **Qwen closure and source-context effects remain small and mixed, whereas gpt-oss source-context estimates are positive at $B = 256$ and 512.** All rows use candidate-logit scoring. Panel a compares three forced-answer closures for the same Qwen prompt-minus-surprise contrast. Panel a pairs all 198 GPQA Diamond and 500 MMLU-Pro items, with three runs per item, and reports paired item-bootstrap intervals. Panel b reconstructs each source prompt at the branch point. Qwen rows are numeric/concision prompt minus surprise stop; gpt-oss rows are lower source effort minus high source effort. Deltas are percentage points with paired item-bootstrap intervals. The Qwen results cover all four horizons, whereas the gpt-oss results cover $B = 256$ and 512.

a Forced-answer closure wording for Qwen3-14B

Benchmark	B	Limited-time Δ [95% CI]	Paraphrase Δ [95% CI]	Neutral Δ [95% CI]	Neutral minus limited-time
GPQA Diamond	512	-1.3 $[-4.5, +1.7]$	-1.7 $[-4.7, +1.3]$	-0.8 $[-4.9, +3.0]$	$+0.5$ $[-4.0, +5.1]$
GPQA Diamond	2,048	-1.3 $[-4.5, +1.7]$	-1.0 $[-4.0, +2.0]$	-0.8 $[-4.5, +2.9]$	$+0.5$ $[-1.7, +2.7]$
GPQA Diamond	8,192	-2.4 $[-5.4, +0.7]$	-1.9 $[-4.7, +1.2]$	-2.9 $[-5.9, +0.2]$	-0.5 $[-2.2, +1.2]$
GPQA Diamond	16,384	$+1.6$ $[-1.8, +5.2]$	$+2.1$ $[-1.3, +5.7]$	$+1.8$ $[-1.7, +5.4]$	$+0.2$ $[+0.0, +0.5]$
MMLU-Pro	512	$+1.7$ $[+0.0, +3.5]$	$+2.4$ $[+0.7, +4.1]$	$+1.9$ $[+0.1, +3.7]$	$+0.1$ $[-0.8, +1.1]$
MMLU-Pro	2,048	$+0.9$ $[-0.7, +2.4]$	$+0.8$ $[-0.8, +2.4]$	$+1.0$ $[-0.5, +2.5]$	$+0.1$ $[-0.3, +0.6]$
MMLU-Pro	8,192	-0.2 $[-1.5, +1.2]$	-0.1 $[-1.5, +1.3]$	-0.2 $[-1.5, +1.2]$	$+0.0$ $[-0.2, +0.2]$
MMLU-Pro	16,384	-0.7 $[-2.2, +0.6]$	-0.7 $[-2.1, +0.7]$	-0.6 $[-2.1, +0.7]$	$+0.1$ $[+0.0, +0.3]$

b Source-context branch readout

Model	Benchmark	B	Contrast	n	Arm A acc.	Arm B acc.	Δ [95% CI]
Qwen3-14B	GPQA Diamond	512	prompt–surprise	198	40.4%	44.9%	-4.5 $[-9.3, 0.0]$
Qwen3-14B	GPQA Diamond	2,048	prompt–surprise	198	53.4%	55.6%	-2.2 $[-5.7, +1.3]$
Qwen3-14B	GPQA Diamond	8,192	prompt–surprise	198	59.1%	61.8%	-2.7 $[-5.7, +0.5]$
Qwen3-14B	GPQA Diamond	16,384	prompt–surprise	198	64.1%	62.3%	$+1.9$ $[-1.5, +5.4]$
Qwen3-14B	MMLU-Pro	512	prompt–surprise	500	60.7%	58.7%	$+1.9$ $[+0.1, +3.8]$
Qwen3-14B	MMLU-Pro	2,048	prompt–surprise	500	72.0%	70.9%	$+1.1$ $[-0.5, +2.6]$
Qwen3-14B	MMLU-Pro	8,192	prompt–surprise	500	75.9%	76.1%	-0.2 $[-1.5, +1.1]$
Qwen3-14B	MMLU-Pro	16,384	prompt–surprise	500	75.9%	76.5%	-0.6 $[-2.1, +0.7]$
gpt-oss-20b	GPQA Diamond	256	low–high	196	53.4%	39.6%	$+13.8$ $[+6.1, +21.4]$
gpt-oss-20b	GPQA Diamond	512	low–high	196	57.0%	44.9%	$+12.1$ $[+4.4, +19.7]$
gpt-oss-20b	GPQA Diamond	256	medium–high	196	42.5%	39.6%	$+2.9$ $[-2.4, +8.2]$
gpt-oss-20b	GPQA Diamond	512	medium–high	196	50.0%	44.9%	$+5.1$ $[-0.9, +11.2]$
gpt-oss-20b	MMLU-Pro	256	low–high	500	62.1%	49.6%	$+12.5$ $[+9.1, +16.1]$
gpt-oss-20b	MMLU-Pro	512	low–high	500	63.7%	55.6%	$+8.1$ $[+4.6, +11.7]$
gpt-oss-20b	MMLU-Pro	256	medium–high	500	56.1%	49.6%	$+6.5$ $[+4.0, +9.0]$
gpt-oss-20b	MMLU-Pro	512	medium–high	500	62.9%	55.6%	$+7.3$ $[+4.7, +9.9]$
gpt-oss-120b	GPQA Diamond	256	low–high	198	51.0%	40.6%	$+10.4$ $[+3.0, +17.8]$
gpt-oss-120b	GPQA Diamond	512	low–high	198	61.1%	42.9%	$+18.2$ $[+10.9, +25.4]$
gpt-oss-120b	GPQA Diamond	256	medium–high	198	43.8%	40.6%	$+3.2$ $[-1.2, +7.6]$
gpt-oss-120b	GPQA Diamond	512	medium–high	198	53.0%	42.9%	$+10.1$ $[+4.5, +15.7]$
gpt-oss-120b	MMLU-Pro	256	low–high	499	69.9%	58.1%	$+11.8$ $[+7.9, +15.8]$
gpt-oss-120b	MMLU-Pro	512	low–high	499	72.5%	65.3%	$+7.1$ $[+3.3, +10.8]$
gpt-oss-120b	MMLU-Pro	256	medium–high	500	65.9%	58.1%	$+7.8$ $[+5.3, +10.4]$
gpt-oss-120b	MMLU-Pro	512	medium–high	500	72.0%	65.3%	$+6.7$ $[+3.9, +9.4]$

Table A29: **Candidate-logit and generated-answer readouts both show higher accuracy for completed lower-effort reasoning, with complete or near-complete coverage.** Each row uses token-exact gpt-oss prefixes matched at the lower-effort policy's terminal horizon. Panel a gives absolute candidate-logit arm means under the common medium-effort answer context; every pair eligible under the matched-horizon rule is replayable. Panel b gives absolute greedy-continuation arm means after averaging the three observed replicates within item. Complete/eligible reports pairs with both continuations relative to runnable pairs, and parse coverage requires both answers to parse.

a Candidate-logit readout, all replayable pairs

Model	Benchmark	Terminal effort	Terminal acc.	High-prefix acc.	Replayable / eligible	Nonreplayable
gpt-oss-20b	GPQA Diamond	low	55.9%	41.4%	594/594	0
gpt-oss-20b	GPQA Diamond	medium	64.9%	47.5%	593/593	0
gpt-oss-20b	MMLU-Pro	low	64.2%	43.3%	1,500/1,500	0
gpt-oss-20b	MMLU-Pro	medium	72.4%	54.4%	1,500/1,500	0
gpt-oss-120b	GPQA Diamond	low	65.0%	41.4%	594/594	0
gpt-oss-120b	GPQA Diamond	medium	68.5%	44.9%	594/594	0
gpt-oss-120b	MMLU-Pro	low	75.5%	49.2%	1,500/1,500	0
gpt-oss-120b	MMLU-Pro	medium	79.1%	54.9%	1,500/1,500	0

b Greedy generated-answer continuation

Model	Benchmark	Terminal effort	Terminal acc.	High-prefix acc.	Complete / eligible	Parse coverage
gpt-oss-20b	GPQA Diamond	low	55.9%	40.4%	594/594	99.7%
gpt-oss-20b	GPQA Diamond	medium	64.7%	47.3%	593/593	98.8%
gpt-oss-20b	MMLU-Pro	low	64.1%	43.7%	1,500/1,500	99.3%
gpt-oss-20b	MMLU-Pro	medium	72.3%	54.1%	1,500/1,500	99.1%
gpt-oss-120b	GPQA Diamond	low	65.0%	41.9%	594/594	99.8%
gpt-oss-120b	GPQA Diamond	medium	68.5%	45.5%	594/594	100.0%
gpt-oss-120b	MMLU-Pro	low	75.5%	49.5%	1,500/1,500	99.7%
gpt-oss-120b	MMLU-Pro	medium	79.1%	55.0%	1,500/1,500	99.5%

Table A30: **Correct-option probabilities are markedly lower among incorrect completed lower-effort predictions than among incorrect matched-horizon high-effort predictions.** Rows condition separately within each arm on an incorrect candidate-readout answer among all replayable token-exact pairs. Mean conditional log loss and the share with option-normalized $q_{\text{gold}} < 10^{-4}$ describe the lower tail of correct-option probability. Counts are arm-specific incorrect item–replicate probes and need not match across arms. Means first average observed replicates within item and then average items.

Model	Benchmark	Effort	Arm	Incorrect probes	Mean log loss	$q_{\text{gold}} < 10^{-4}$
gpt-oss-20b	GPQA Diamond	low	Terminal	262	12.745	80.1%
			High prefix	348	1.936	0.3%
gpt-oss-20b	GPQA Diamond	medium	Terminal	208	16.558	98.4%
			High prefix	312	2.952	3.8%
gpt-oss-20b	MMLU-Pro	low	Terminal	537	14.657	88.9%
			High prefix	851	2.862	0.8%
gpt-oss-20b	MMLU-Pro	medium	Terminal	414	18.315	99.1%
			High prefix	684	4.460	11.1%
gpt-oss-120b	GPQA Diamond	low	Terminal	208	25.289	99.6%
			High prefix	348	5.233	12.4%
gpt-oss-120b	GPQA Diamond	medium	Terminal	187	26.860	100.0%
			High prefix	327	5.502	14.5%
gpt-oss-120b	MMLU-Pro	low	Terminal	368	25.343	99.4%
			High prefix	762	5.486	15.4%
gpt-oss-120b	MMLU-Pro	medium	Terminal	314	26.839	100.0%
			High prefix	677	6.658	24.8%

Table A31: **In single-rollout Qwen3-8B and Omni-MATH-2 comparisons, Qwen3-8B shows no broad prompt gain and every gpt-oss fixed-checkpoint difference is positive.** Panel a compares one Qwen3-8B numeric/concision-prompt rollout with one separately generated surprise-stop rollout; the accuracy denominator is the number of unique paired items. For MMLU-Pro at $B = 2{,}048$, the length medians use 499 paired item–trace records although the accuracy contrast contains 498 items. Panel b starts from a difficulty-stratified 500-item Omni-MATH-2 draw; the analyzed model-specific paired cohorts contain 475 items for gpt-oss-20b and 490 for gpt-oss-120b. The remaining 25 and 10 sampled items are those whose high-effort trace produced no parseable final answer; all 25 for gpt-oss-20b and 1 of the 10 for gpt-oss-120b had reached the generation limit. Panel b uses one rollout per effort setting, with GPT-5 mini judging free-form answers at common source-token checkpoints. Intervals bootstrap items. These estimates characterize the sampled trajectories and do not quantify variation across generated replicates.

a Qwen3-8B single-rollout comparison

Benchmark	B	n	Prompt acc.	Surprise acc.	Δ [95% CI]	Sign-flip p	Median prompt L	Median surprise L
GPQA Diamond	512	196	34.2%	36.7%	-2.6 $[-6.1, +1.0]$	0.269	5,022	5,929
GPQA Diamond	2,048	198	43.4%	42.9%	$+0.5$ $[-4.5, +5.6]$	1.000	5,096	5,957
MMLU-Pro	512	498	53.4%	52.6%	$+0.8$ $[-1.8, +3.2]$	0.638	1,286	1,623
MMLU-Pro	2,048	498	67.7%	65.1%	$+2.6$ $[-0.2, +5.2]$	0.090	1,247	1,639

b gpt-oss fixed-checkpoint Omni-MATH-2 readout

Model	Contrast	B	n	Lower acc.	High acc.	Δ [95% CI]
gpt-oss-20b	low–high	512	475	29.9%	18.3%	$+11.6$ $[+8.4, +14.9]$
gpt-oss-20b	low–high	1,024	475	38.1%	21.7%	$+16.4$ $[+12.6, +20.2]$
gpt-oss-20b	low–high	2,048	475	45.9%	29.1%	$+16.8$ $[+12.6, +20.8]$
gpt-oss-20b	low–high	4,096	475	48.2%	34.9%	$+13.3$ $[+9.3, +17.3]$
gpt-oss-20b	medium–high	512	475	21.7%	18.3%	$+3.4$ $[+0.8, +6.1]$
gpt-oss-20b	medium–high	1,024	475	25.9%	21.7%	$+4.2$ $[+1.3, +7.4]$
gpt-oss-20b	medium–high	2,048	475	38.7%	29.1%	$+9.7$ $[+5.9, +13.5]$
gpt-oss-20b	medium–high	4,096	475	46.1%	34.9%	$+11.2$ $[+7.4, +14.9]$
gpt-oss-120b	low–high	512	490	39.0%	26.9%	$+12.0$ $[+8.2, +15.9]$
gpt-oss-120b	low–high	1,024	490	50.4%	30.6%	$+19.8$ $[+15.9, +23.9]$
gpt-oss-120b	low–high	2,048	490	59.0%	41.0%	$+18.0$ $[+13.7, +22.2]$
gpt-oss-120b	low–high	4,096	490	60.6%	52.7%	$+8.0$ $[+3.9, +12.0]$
gpt-oss-120b	medium–high	512	490	30.0%	26.9%	$+3.1$ $[+0.0, +6.1]$
gpt-oss-120b	medium–high	1,024	490	39.6%	30.6%	$+9.0$ $[+5.5, +12.4]$
gpt-oss-120b	medium–high	2,048	490	53.9%	41.0%	$+12.9$ $[+9.2, +16.5]$
gpt-oss-120b	medium–high	4,096	490	65.5%	52.7%	$+12.9$ $[+9.0, +16.7]$

Table A32: **The Qwen MMLU-Pro $B = 512$ contrast varies by category, and a text-reconstructed four-horizon concise/early-answer comparison shows no broad advantage.** Panel a pools the main 500-item subset and an independent 500-item subset at $B = 512$. Accuracies average three rollout-level candidate-logit readouts within item and then average items within category; no category-level intervals are available. In panel a, prompt denotes the numeric/concision prompt and Δ is prompt minus surprise stop. Panel b compares the concise/early-answer prompt with independently generated surprise-stop and numeric/concision-prompt cohorts; its deltas are concise/early-answer minus the named comparator. Both panels use cohorts analysed from decoded text rather than stored token IDs. Panel a describes category variation across two subsets and panel b extends the prompt comparison across four horizons, but neither contributes to the token-exact main estimates. Panel b intervals average three stored runs within item and bootstrap items. Deltas are reported in percentage points.

a MMLU-Pro category decomposition at $B = 512$

Category	n	Surprise acc.	Prompt acc.	Δ (pp)
Biology	60	77.8%	80.0%	+2.2
Business	66	60.1%	63.1%	+3.0
Chemistry	94	51.8%	57.4%	+5.7
Computer science	34	64.7%	72.5%	+7.8
Economics	70	72.4%	76.7%	+4.3
Engineering	80	43.8%	42.9%	−0.8
Health	68	68.6%	66.7%	−2.0
History	32	69.8%	67.7%	−2.1
Law	92	31.5%	37.0%	+5.4
Math	112	58.9%	60.4%	+1.5
Other	76	54.4%	53.9%	−0.4
Philosophy	42	61.1%	54.0%	−7.1
Physics	108	50.9%	55.2%	+4.3
Psychology	66	72.2%	74.7%	+2.5
Pooled total	1,000	57.6%	59.8%	+2.1

b Four-horizon concise/early-answer prompt comparison

Benchmark	B	Concise acc.	Surprise acc.	Δ vs. surprise [95% CI]	Numeric-prompt acc.	Δ vs. numeric prompt [95% CI]
GPQA Diamond	512	43.1%	42.9%	+0.2 [−4.0, +4.4]	43.9%	−0.8 [−3.5, +1.9]
GPQA Diamond	2,048	56.1%	54.9%	+1.2 [−2.5, +4.7]	53.0%	+3.0 [+0.0, +6.2]
GPQA Diamond	8,192	59.8%	61.8%	−2.0 [−5.4, +1.3]	58.9%	+0.8 [−1.9, +3.5]
GPQA Diamond	16,384	60.8%	62.3%	−1.5 [−4.9, +1.7]	64.1%	−3.4 [−6.7, −0.2]
MMLU-Pro	512	63.5%	58.7%	+4.9 [+2.8, +7.0]	60.9%	+2.7 [+0.9, +4.5]
MMLU-Pro	2,048	71.8%	70.8%	+1.0 [−0.7, +2.7]	71.9%	−0.1 [−1.7, +1.5]
MMLU-Pro	8,192	74.1%	76.1%	−2.1 [−3.5, −0.7]	75.9%	−1.9 [−3.3, −0.4]
MMLU-Pro	16,384	73.9%	76.5%	−2.7 [−4.1, −1.3]	75.9%	−2.0 [−3.4, −0.6]

Table A33: **An independent MMLU-Pro subset preserves the small numeric-prompt gain at 512 tokens but not at 2,048 tokens.** The original-subset row uses the 500-item MMLU-Pro sample corresponding to the design in Table A13; it remains a separate replication rather than part of the token-exact main analysis. The independent row uses a second category-stratified 500-item subset that excludes the original items, and the pooled row combines both subsets. At $B = 2{,}048$, the numeric/concision and concise/early-answer instructions are compared with the same independent-subset surprise-stop baseline under the same three-run procedure. Accuracy and survival use the candidate-logit readout over three stored Qwen3-14B runs. Intervals are paired two-way item $\times$ run bootstrap intervals. The $B = 512$ rows use the text-reconstructed records and three stored runs of these cohorts, so their intervals do not support generalization across generation seeds; the $B = 2{,}048$ rows use stored completion token IDs and recorded seeds. Per-item paired outcomes are retained for every row.

Sample and instruction	B	n	Prompt acc.	Surprise acc.	Δ [95% CI]	$P(L > B)$ prompt	$P(L > B)$ surprise	Median prompt L	Median surprise L
Original 500, numeric/concision	512	500	60.4%	58.7%	+1.7 [−0.6, +4.0]	75.9%	81.3%	876	1,184
Independent 500, numeric/concision	512	500	59.1%	56.6%	+2.5 [+0.4, +4.8]	77.1%	81.5%	1,048	1,314
Pooled 1,000, numeric/concision	512	1,000	59.8%	57.6%	+2.1 [+0.6, +3.6]	76.5%	81.4%	955	1,214
Independent 500, numeric/concision	2,048	500	71.5%	71.7%	−0.2 [−2.6, +2.2]	26.3%	36.0%	1,043	1,314
Independent 500, concise/early-answer	2,048	500	74.4%	71.7%	+2.7 [+0.5, +5.0]	19.9%	36.0%	812	1,314

Table A34: **Qwen prompt controls change reasoning length more consistently than accuracy at the 512-token stop.** Each row compares a Qwen3-14B instruction with the same surprise-stop baseline at $B = 512$, using three stored runs averaged within item. The accuracy columns report terminal-aware policy accuracy, with candidate-logit scoring for unfinished prefixes, and the displayed median is the full reasoning length under the instruction. These cohorts were analysed from decoded text rather than stored token IDs and therefore characterize prompt patterns descriptively. The token-exact paired candidate-logit comparison is reported in Table A14. Accuracy differences are instruction minus surprise stop in percentage points, with paired item-cluster bootstrap intervals. The four-horizon concise/early-answer results and direct prompt-to-prompt contrasts appear in Table A32. Full survival, survivor-accuracy, terminal-aware accuracy, and $\Delta \log L$ results are included in the companion data release.

Instruction	Benchmark	n	Instruction acc.	Surprise acc.	Δ (pp)	95% CI	Median instruction L
Numeric/concision	GPQA Diamond	198	43.9%	42.9%	+1.0	[−2.5, +4.5]	3,641
Limited, no number	GPQA Diamond	198	45.3%	42.9%	+2.4	[−1.7, +6.4]	3,654
Number only	GPQA Diamond	198	43.8%	42.9%	+0.8	[−3.2, +4.9]	3,690
Advance exact-stop notice	GPQA Diamond	198	43.6%	42.9%	+0.7	[−3.9, +5.1]	3,690
Approximate budget	GPQA Diamond	198	41.4%	42.9%	−1.5	[−5.4, +2.2]	3,904
Irrelevant number	GPQA Diamond	198	42.4%	42.9%	−0.5	[−4.7, +3.5]	4,222
Concise/early-answer	GPQA Diamond	198	43.1%	42.9%	+0.2	[−4.0, +4.4]	3,080
Thorough semantic	GPQA Diamond	198	42.6%	42.9%	−0.3	[−4.4, +3.7]	4,847
Numeric/concision	MMLU-Pro	500	60.9%	58.7%	+2.2	[+0.3, +4.1]	876
Limited, no number	MMLU-Pro	500	61.0%	58.7%	+2.3	[+0.5, +4.1]	892
Number only	MMLU-Pro	500	60.9%	58.7%	+2.3	[+0.5, +4.0]	912
Advance exact-stop notice	MMLU-Pro	500	59.9%	58.7%	+1.3	[−0.5, +3.0]	880
Approximate budget	MMLU-Pro	500	60.9%	58.7%	+2.2	[+0.5, +4.0]	938
Irrelevant number	MMLU-Pro	500	59.9%	58.7%	+1.2	[−0.5, +3.0]	1,065
Concise/early-answer	MMLU-Pro	500	63.5%	58.7%	+4.9	[+2.8, +7.0]	710
Thorough semantic	MMLU-Pro	500	59.7%	58.7%	+1.0	[−0.7, +2.7]	1,299

Table A35: **Continuing a Qwen prefix from a trace with an incorrect natural final answer can rescue the answer, but anchoring remains substantial.** Each cell reports the rescue rate and anchoring rate in percent after replaying the first B tokens of a Qwen3-14B natural trace whose final answer is incorrect. The analysis uses the returned-top-20 forced-answer readout rather than full-vocabulary candidate-logit scoring. "Answer now" closes reasoning immediately. The continuation conditions request $R = 32{,}768$ additional reasoning tokens before the forced answer, with the remaining budget either hidden or stated in the prompt. The effective maximum is limited by the context window in every displayed continuation cell, with medians from approximately $24{,}100$ to $31{,}900$ tokens.

Benchmark	B	n	Rescue / anchoring rate (%)		
			Answer now	Continue, hidden R	Continue, visible R
GPQA Diamond	512	123	30.1/40.7	38.2/37.4	37.4/35.8
GPQA Diamond	2,048	112	24.1/45.5	33.9/42.9	27.7/50.0
GPQA Diamond	8,192	34	20.6/55.9	26.5/47.1	26.5/47.1
MMLU-Pro	512	176	26.1/43.2	31.2/51.1	31.2/51.7
MMLU-Pro	2,048	95	20.0/55.8	29.5/48.4	29.5/55.8
MMLU-Pro	8,192	30	26.7/46.7	33.3/50.0	30.0/50.0

Table A36: **The prompted budget is decodable at the prompt boundary but mostly falls to label-shuffle levels after reasoning begins.** The table reports the best-layer activation readouts for Qwen3-14B at representative token positions k. "Bundled prompt?" is a binary readout over surprise-stop and numeric/concision-prompt traces. "Which budget?" is a four-class readout over numeric/concision-prompt traces only, with $B \in \{512, 2{,}048, 8{,}192, 16{,}384\}$. Balanced accuracy uses item-grouped cross-validation. The shuffle column is the 95th percentile over 20 label shuffles.

Readout	Benchmark	k	Best layer	Items	Balanced acc.	Shuffle q95
Bundled prompt?	GPQA Diamond	0	10	197	1.000	0.546
Bundled prompt?	GPQA Diamond	64	39	197	0.621	0.533
Bundled prompt?	GPQA Diamond	512	39	197	0.700	0.526
Bundled prompt?	GPQA Diamond	2,048	39	163	0.623	0.539
Bundled prompt?	MMLU-Pro	0	10	200	1.000	0.537
Bundled prompt?	MMLU-Pro	64	39	200	0.627	0.540
Bundled prompt?	MMLU-Pro	512	39	178	0.689	0.548
Bundled prompt?	MMLU-Pro	2,048	39	75	0.588	0.580
Which budget?	GPQA Diamond	0	39	197	0.938	0.283
Which budget?	GPQA Diamond	64	10	197	0.258	0.268
Which budget?	GPQA Diamond	512	30	197	0.278	0.269
Which budget?	GPQA Diamond	2,048	39	156	0.262	0.286
Which budget?	MMLU-Pro	0	39	200	0.905	0.278
Which budget?	MMLU-Pro	64	0	200	0.261	0.279
Which budget?	MMLU-Pro	512	20	171	0.259	0.278
Which budget?	MMLU-Pro	2,048	0	69	0.343	0.301

Table A37: **Matching the stated number to the actual stop barely changes when Qwen3-14B finishes. a**, Share of runs whose reasoning finished naturally before the imposed stop ("Finished") and the share that finished with a correct answer, for the six instructions of the prompt-selection cohort, the 500-item MMLU-Pro subset with eight paired replicates per item. Differences are instruction minus no advance notice in percentage points. "Finished and correct" scores the completed block's fresh $R = 64$ source-policy answer, as in Figure 4. The candidate-logit correct-completion values in Table A15 differ slightly. **b**, Matching contrast within each prompt family, defined as half of $H = 512$ minus $H = 2{,}048$ at $B = 512$ plus $H = 2{,}048$ minus $H = 512$ at $B = 2{,}048$. Intervals are pointwise 95% nested item–replicate bootstrap intervals from 10,000 draws.

a Completion before the imposed stop

Imposed stop B	Prompt instruction	Number stated H	Finished	Δ finished [95% CI]	Δ finished and correct [95% CI]
512	No advance notice	—	20.3%	—	—
	Numeric/concision prompt	512	25.0%	+4.6 [+3.1, +6.3]	+4.3 [+2.9, +5.9]
	Numeric/concision prompt	2,048	23.5%	+3.2 [+1.7, +4.8]	+3.0 [+1.6, +4.4]
	Advance notice of exact stop	512	25.4%	+5.1 [+3.6, +6.6]	+4.5 [+3.0, +6.0]
	Advance notice of exact stop	2,048	23.0%	+2.6 [+1.1, +4.2]	+2.6 [+1.2, +4.1]
	Concise/early-answer instruction	—	36.4%	+16.1 [+13.8, +18.4]	+13.8 [+11.6, +16.0]
2,048	No advance notice	—	72.1%	—	—
	Numeric/concision prompt	512	77.6%	+5.5 [+3.9, +7.1]	+4.4 [+2.8, +6.1]
	Numeric/concision prompt	2,048	77.1%	+5.0 [+3.4, +6.7]	+4.0 [+2.5, +5.6]
	Advance notice of exact stop	512	78.4%	+6.3 [+4.6, +7.9]	+5.2 [+3.6, +6.9]
	Advance notice of exact stop	2,048	77.3%	+5.2 [+3.6, +6.8]	+4.6 [+3.1, +6.3]
	Concise/early-answer instruction	—	82.3%	+10.1 [+8.2, +12.2]	+8.2 [+6.3, +10.1]

b Direct test of matching the stated number to the stop

Prompt instruction	Outcome	Match advantage (pp)	95% CI
Numeric/concision prompt	Finished	+0.5	[−0.4, +1.4]
Numeric/concision prompt	Finished and correct	+0.5	[−0.3, +1.3]
Advance notice of exact stop	Finished	+0.7	[−0.2, +1.6]
Advance notice of exact stop	Finished and correct	+0.6	[−0.2, +1.5]

Table A38: **The concise instruction's advantage persists among pairs in which both answers parse.** Qwen3-14B in the prompt-selection cohort, the 500-item MMLU-Pro subset with eight paired generations per instruction and fresh $R = 64$ answers. Rows condition on item–replicate pairs whose ordinary and concise answers both parse, so they describe a selected population and exclude the usable-answer component of the full-cohort difference in Figure 4. "All pairs" repeats the full-cohort difference with parsing failures counted as incorrect. Differences are concise minus ordinary in percentage points. Intervals are pointwise 95% nested item–replicate bootstrap intervals of the conditional difference from 10,000 draws.

Answer context	B	Both parsed	Share	Ordinary	Concise	Δ both parsed [95% CI]	Δ all pairs
Source-policy context	512	3,766	94.2%	62.9%	66.4%	+3.5 [+1.7, +5.3]	+3.9
Source-policy context	2,048	3,855	96.4%	73.5%	74.2%	+0.7 [−0.7, +2.1]	+2.1
Common replay context	512	3,758	94.0%	62.9%	66.0%	+3.0 [+1.4, +4.8]	+3.3
Common replay context	2,048	3,840	96.0%	73.6%	74.4%	+0.8 [−0.6, +2.1]	+1.9

Table A39: **A replication on MMLU-Pro questions not used to choose the prompts reproduces the concise-instruction pattern and the null matching contrast.** Qwen3-14B on a fresh category-stratified 500-item MMLU-Pro subset disjoint from the prompt-selection subset and the text-reconstructed independent subset, with 8 paired generations per instruction, the same instructions and parser, and fresh $R = 64$ greedy answers for completed and interrupted reasoning blocks. **a**, Source-policy context. "Finished" is the share of runs whose reasoning ended naturally before the imposed stop. Differences are instruction minus no advance notice in percentage points with parsing failures counted as incorrect. **b**, Concise-minus-ordinary differences under the common replay context, the same difference among pairs in which neither trace finished before the stop, and the matching contrasts within each numeric prompt family, defined as in Figure 4c. **c**, Sensitivity to answer handling. Completed reasoning blocks keep their natural answer instead of receiving a fresh one, and the unfinished-pair contrast is restricted to pairs whose two answers both parse. **d**, Natural completion without an imposed stop: accuracy of the parsed final answer with failures counted as incorrect, generated tokens including reasoning and answer, and paired instruction-minus-ordinary differences. Intervals are pointwise 95% nested item–replicate bootstrap intervals from 10,000 draws with seed 0.

a Generated answers in the source-policy context

		Imposed stop $B = 512$			Imposed stop $B = 2{,}048$		
Prompt instruction	Number stated H	Finished	Accuracy	Δ [95% CI]	Finished	Accuracy	Δ [95% CI]
No advance notice	—	16.9%	59.6%	—	70.1%	70.7%	—
Numeric/concision prompt	512	20.7%	61.0%	+1.4 [+0.0, +2.8]	76.1%	72.1%	+1.4 [+0.0, +2.8]
Numeric/concision prompt	2,048	19.2%	60.3%	+0.7 [−0.9, +2.2]	75.1%	72.0%	+1.3 [−0.1, +2.8]
Advance notice of exact stop	512	21.7%	61.9%	+2.2 [+0.7, +3.7]	77.1%	72.8%	+2.1 [+0.7, +3.5]
Advance notice of exact stop	2,048	20.1%	61.8%	+2.2 [+0.8, +3.6]	75.7%	72.3%	+1.5 [+0.2, +3.0]
Concise/early-answer instruction	—	34.0%	63.1%	+3.5 [+1.8, +5.2]	80.9%	73.2%	+2.5 [+0.9, +4.1]

b Common replay context and matching contrasts

Contrast	Answer context	Δ [95% CI]
Concise/early-answer minus ordinary, $B = 512$	Common replay context	+3.9 [+2.4, +5.6]
Concise/early-answer minus ordinary, $B = 2{,}048$	Common replay context	+1.9 [+0.4, +3.4]
Concise/early-answer minus ordinary, both unfinished at $B = 512$ (2,562 pairs)	Source-policy context	+4.1 [+1.7, +6.4]
Concise/early-answer minus ordinary, both unfinished at $B = 512$ (2,562 pairs)	Common replay context	+4.7 [+2.6, +6.9]
Concise/early-answer minus ordinary, both unfinished at $B = 2{,}048$ (716 pairs)	Source-policy context	+8.5 [+3.8, +13.6]
Concise/early-answer minus ordinary, both unfinished at $B = 2{,}048$ (716 pairs)	Common replay context	+5.3 [+1.1, +9.7]
Numeric/concision prompt match advantage	Source-policy context	+0.3 [−0.5, +1.1]
Numeric/concision prompt match advantage	Common replay context	+0.1 [−0.7, +0.9]
Advance notice of exact stop match advantage	Source-policy context	−0.2 [−1.0, +0.5]
Advance notice of exact stop match advantage	Common replay context	−0.1 [−0.9, +0.6]

c Sensitivity to answer handling

Contrast	Answer context	Δ [95% CI]
Concise/early-answer minus ordinary, natural answers retained, $B = 512$	Source-policy context	+3.5 [+1.8, +5.2]
Concise/early-answer minus ordinary, natural answers retained, $B = 2{,}048$	Source-policy context	+1.8 [+0.4, +3.2]
Numeric/concision prompt match advantage, natural answers retained	Source-policy context	+0.3 [−0.4, +1.1]
Advance notice of exact stop match advantage, natural answers retained	Source-policy context	−0.3 [−1.0, +0.5]
Concise/early-answer minus ordinary, both unfinished and both parsed at $B = 512$ (2,341 pairs)	Source-policy context	+3.5 [+1.2, +5.9]
Concise/early-answer minus ordinary, both unfinished and both parsed at $B = 512$ (2,325 pairs)	Common replay context	+4.2 [+2.0, +6.4]
Concise/early-answer minus ordinary, both unfinished and both parsed at $B = 2{,}048$ (591 pairs)	Source-policy context	+3.2 [−0.7, +7.3]
Concise/early-answer minus ordinary, both unfinished and both parsed at $B = 2{,}048$ (569 pairs)	Common replay context	+1.9 [−2.0, +6.0]

d Natural completion

Prompt instruction	Accuracy	Δ accuracy [95% CI]	Mean tokens	Median	Δ mean tokens [95% CI]
No advance notice	76.6%	—	2,358	1,124	—
Numeric/concision prompt, $H = 512$	75.8%	−0.8 [−1.9, +0.4]	1,924	947	−434 [−526, −349]
Numeric/concision prompt, $H = 2{,}048$	75.1%	−1.5 [−2.7, −0.4]	1,959	959	−399 [−489, −319]
Advance notice of exact stop, $H = 512$	75.9%	−0.7 [−1.9, +0.5]	1,870	920	−489 [−579, −402]
Advance notice of exact stop, $H = 2{,}048$	75.8%	−0.8 [−2.0, +0.3]	1,973	938	−386 [−468, −311]
Concise/early-answer instruction	75.6%	−1.0 [−2.3, +0.2]	1,626	726	−732 [−833, −639]

Table A40: **The replication's jointly unfinished concise advantage is stable under within-item re-pairing.** Each of 10,000 draws independently permutes the eight concise-instruction replicates against the eight ordinary replicates within every item of the replication cohort, recomputes which pairs are jointly unfinished at B, and re-estimates the concise-minus-ordinary generated-answer difference among them. The original pairing shares the generation seed across instructions. Ranges are sensitivity ranges over re-pairings, not sampling intervals.

Answer context	B	Original pairs	Original Δ	Re-pairing range	Share positive
Source-policy context	512	2,562	+4.1	[+3.5, +4.6]	100.0%
Source-policy context	2,048	716	+8.5	[+6.1, +9.5]	100.0%
Common replay context	512	2,562	+4.7	[+4.1, +5.3]	100.0%
Common replay context	2,048	716	+5.3	[+3.1, +6.4]	100.0%

Table A41: **Adding the announced number to the concise instruction does not consistently improve its early answers.** Four further instructions were generated on the replication's 500 MMLU-Pro items with the same 8 paired generation seeds as its six instructions: the concise instruction followed by the advance notice of an exact stop or by the numeric/concision prompt, announcing 512 or 2,048 tokens. Every capped reasoning block receives a fresh greedy answer with $R = 64$, and parsing failures count as incorrect. **a**, Accuracy and share of reasoning blocks completed by each horizon in the source-policy context. **b**, Differences at the announced horizon from the concise instruction, from ordinary generation, and from the same budget instruction without the concise sentence. **c**, Matching contrasts $\frac{1}{2}[(\Delta_{512} - \Delta_{2,048})$ at $512 + (\Delta_{2,048} - \Delta_{512})$ at $2{,}048]$ within each concise-plus-number family, where Δ_h denotes the instruction announcing h tokens. Intervals are pointwise 95% nested item–replicate bootstrap intervals (10,000 draws, seed 0).

a, Accuracy (%) and completion by the horizon, source-policy context

Instruction	Acc. 512	Finished 512	Acc. 2,048	Finished 2,048
No advance notice	59.6	16.9%	70.7	70.1%
Numeric/concision prompt, 512	61.0	20.7%	72.1	76.1%
Numeric/concision prompt, 2,048	60.3	19.2%	72.0	75.1%
Advance notice of exact stop, 512	61.9	21.7%	72.8	77.1%
Advance notice of exact stop, 2,048	61.8	20.1%	72.3	75.7%
Concise/early-answer instruction	63.1	34.0%	73.2	80.9%
Concise + exact-stop notice, 512	64.2	38.7%	73.6	84.4%
Concise + exact-stop notice, 2,048	63.6	34.7%	73.1	83.1%
Concise + numeric/concision, 512	60.2	48.6%	66.4	89.1%
Concise + numeric/concision, 2,048	61.4	41.7%	68.8	86.1%

b, Differences at the announced horizon (percentage points, 95% CI)

Instruction	Comparator	Source-policy context	Common replay context
Concise + exact-stop notice, 512	Concise instruction	+1.1 [−0.3, +2.5]	+1.3 [+0.0, +2.6]
Concise + exact-stop notice, 512	Ordinary generation	+4.6 [+2.6, +6.6]	+5.2 [+3.4, +7.1]
Concise + exact-stop notice, 512	Advance notice of exact stop, 512	+2.4 [+0.6, +4.1]	+3.4 [+1.8, +5.0]
Concise + exact-stop notice, 2,048	Concise instruction	−0.1 [−1.3, +1.1]	+0.2 [−1.0, +1.4]
Concise + exact-stop notice, 2,048	Ordinary generation	+2.4 [+0.8, +4.0]	+2.1 [+0.6, +3.7]
Concise + exact-stop notice, 2,048	Advance notice of exact stop, 2,048	+0.9 [−0.5, +2.2]	+1.0 [−0.2, +2.3]
Concise + numeric/concision, 512	Concise instruction	−3.0 [−4.9, −1.0]	−2.6 [−4.4, −0.8]
Concise + numeric/concision, 512	Ordinary generation	+0.6 [−1.6, +2.6]	+1.4 [−0.6, +3.3]
Concise + numeric/concision, 512	Numeric/concision prompt, 512	−0.8 [−2.7, +1.1]	−0.1 [−1.9, +1.7]
Concise + numeric/concision, 2,048	Concise instruction	−4.4 [−6.1, −2.7]	−3.3 [−4.9, −1.7]
Concise + numeric/concision, 2,048	Ordinary generation	−1.9 [−3.9, +0.1]	−1.4 [−3.3, +0.6]
Concise + numeric/concision, 2,048	Numeric/concision prompt, 2,048	−3.2 [−5.0, −1.5]	−2.3 [−4.1, −0.6]

c, Matching contrasts within each family (percentage points, 95% CI)

Family	Source-policy context	Common replay context
Concise + exact-stop notice	+0.1 [−0.6, +0.8]	+0.2 [−0.5, +0.9]
Concise + numeric/concision	+0.6 [−0.1, +1.4]	+0.5 [−0.2, +1.2]

Table A42: **Adding numeric wording to gpt-oss effort produces no broad advantage at the 512-token stop.** Each row compares a numeric-budget instruction crossed with the indicated effort setting against the effort-only candidate-logit baseline at the same $B = 512$ stop. Differences are numeric-instruction minus effort-only accuracy in percentage points, with paired bootstrap intervals over items. Longer announced-budget contrasts lack a directly matched effort-only baseline because the candidate-logit effort-only comparison for this interaction covers only $B = 512$.

Model	Benchmark	Effort	n	Numeric acc.	Effort-only acc.	Δ (pp)	95% CI
gpt-oss-20b	GPQA Diamond	low	198	55.6%	56.1%	−0.5	[−7.6, +6.1]
gpt-oss-20b	GPQA Diamond	medium	198	44.4%	50.0%	−5.6	[−10.6, −0.5]
gpt-oss-20b	GPQA Diamond	high	196	44.4%	44.4%	+0.0	[−6.1, +6.1]
gpt-oss-20b	MMLU-Pro	low	498	66.5%	63.7%	+2.9	[−0.5, +6.5]
gpt-oss-20b	MMLU-Pro	medium	500	63.0%	63.0%	+0.0	[−2.7, +2.7]
gpt-oss-20b	MMLU-Pro	high	499	58.9%	56.2%	+2.7	[+0.0, +5.5]
gpt-oss-120b	GPQA Diamond	low	198	64.1%	61.1%	+3.0	[−3.5, +9.6]
gpt-oss-120b	GPQA Diamond	medium	198	53.5%	53.0%	+0.5	[−4.2, +5.2]
gpt-oss-120b	GPQA Diamond	high	198	41.4%	43.9%	−2.5	[−7.1, +2.0]
gpt-oss-120b	MMLU-Pro	low	499	73.7%	73.1%	+0.6	[−2.2, +3.4]
gpt-oss-120b	MMLU-Pro	medium	500	73.2%	71.8%	+1.4	[−0.8, +3.7]
gpt-oss-120b	MMLU-Pro	high	500	66.0%	66.7%	−0.7	[−3.0, +1.6]

Table A43: **The scripted positive control detects a known early-answer policy.** We construct 96 balanced synthetic four-choice items whose questions contain no answer information. The long policy uses a neutral reasoning prefix whose candidate answer appears after 64 tokens, whereas the short policy states the candidate answer immediately. Both policies reveal the candidate answer by $B = 512$. Accuracy uses the Qwen3-14B candidate-logit readout. Differences are short minus long in percentage points, with paired bootstrap intervals over items. The reported p values use a two-sided paired item-level sign-flip randomization test with 10,000 draws.

B	n	Long acc.	Short acc.	Δ (pp)	95% CI	Long reveal	Short reveal	Sign-flip p
64	96	25.0%	100.0%	+75.0	[+65.6, +83.3]	0.0%	100.0%	0.0001
512	96	100.0%	100.0%	+0.0	[+0.0, +0.0]	100.0%	100.0%	1.0000

Table A44: **None of the 2,788 terminal gpt-oss replay prefixes in a separate single-run cohort contains final-channel answer content.** This terminal-replay validation does not exhaustively cover the three-replicate main cohort. It covers terminal low- and medium-effort prefixes in the single-run common-context readout. Answer assertions inside the reasoning text are part of the measured behavior and are not counted as leaks. A final-channel leak is a replay prefix containing a final-channel delimiter. A final-reply suffix leak is the parsed natural final reply appended to the replay prefix when the reply contains at least three non-whitespace characters.

Model	Benchmark	Prefix source	Prefixes	Final-channel leaks	Reply checks	Final-reply suffix leaks
gpt-oss-120b	GPQA Diamond	low terminal	198	0	0	0
gpt-oss-120b	GPQA Diamond	medium terminal	198	0	0	0
gpt-oss-120b	MMLU-Pro	low terminal	500	0	0	0
gpt-oss-120b	MMLU-Pro	medium terminal	500	0	0	0
gpt-oss-20b	GPQA Diamond	low terminal	196	0	0	0
gpt-oss-20b	GPQA Diamond	medium terminal	196	0	0	0
gpt-oss-20b	MMLU-Pro	low terminal	500	0	1	0
gpt-oss-20b	MMLU-Pro	medium terminal	500	0	1	0
Total			2,788	0	2	0