# AI IS BECOMING AN EXPERT REASONER

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

## ABSTRACT

Across physics and chess experiments, the expertise literature shows that what differentiates experts from novices is not more search but less: experts take the necessary steps while novices employ a more circuitous trial-and-error approach. We apply this lens to large reasoning models and ask how their token efficiency has changed over time. We introduce the minimal human derivation, the shortest published human solution to a problem, as a problem-specific benchmark against which excess reasoning can be measured across problems and time. On 40 problems from the 2026 AIME and HMMT competitions, evaluated across 15 frontier models with repeated sampling, mean tokens per correct solution fell 8.5× across OpenAI models (o1 to GPT-6-Astra) and 4.8× across Anthropic releases (Opus 4.5 to Fable 5.1), while accuracy rose in both families. Excess reasoning above the human benchmark falls 22.3–32.6% for OpenAI and 31.7–45.4% for Anthropic per quarter. Extrapolating within family, average frontier traces come within 10% of the human benchmark by late 2027 to late 2028. Matched pairs of open-weight models suggest efficiency gains come from both pre-training scale and post-training. We discuss implications for forecasting, the token economics of AI, and chain-of-thought monitorability as reasoning approaches the length of the shortest human derivation.

## 1 INTRODUCTION

When OpenAI released o1 in December 2024, it solved competition mathematics problems no language model had solved before, but used roughly 23 times as many tokens as the shortest human solution. Twenty-one months later, GPT-6-Astra solves more problems at 2.8 times the human length. This paper measures that compression and how fast the remaining gap is closing.

Large reasoning models (LRMs) trained with reinforcement learning on verifiable rewards (OpenAI, 2024; DeepSeek-AI, 2025; Kimi Team, 2025) now solve problems at the frontier of human capabilities (Nature News, 2026; OpenAI, 2026).

One drawback of early LRMs was their inefficiency: they were prone to overthinking, generating redundant solutions and unnecessarily verifying even on problems as simple as $2 + 3$ (Chen et al., 2024). They used 8× more tokens than non-reasoning models on the same benchmarks (Epoch AI, 2025). Because tokens are the economic unit of language models, unnecessary tokens can create significant costs for both users and providers (de Vries, 2023; Luccioni et al., 2024; Fortune, 2026). Further, since chain of thought is the primary window into a model's reasoning, its length is also a question for oversight (Korbak et al., 2025).

Reasoning efficiency has only recently begun to be measured (Epoch AI, 2025; Du et al., 2025; Potamitis et al., 2025). Most prior work treats efficiency as an engineering target, proposing length penalties and token budgets to compress output and reward concise solutions (Sui et al., 2025; Arora & Zanette, 2025; Aggarwal & Welleck, 2025). Instead, we treat efficiency as an evaluation in itself, borrowing a lens from the study of human expertise. In experiments across domains, what separates experts from novices is not more search but less: experts recognize a problem's structure and take the necessary steps, while novices use trial and error (de Groot, 1978; Chase & Simon, 1973; Chi et al., 1981). We ask the analogous question of language models: how much search does a model require beyond what is minimal for a correct solution, and how quickly is that excess shrinking?

Measuring excess requires knowing the minimum. We introduce the *minimal human derivation* (MHD): the shortest published human solution to a problem, tokenized identically to model out-



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.

put. The MHD is an empirical proxy for the most direct sequence of reasoning steps that reaches a solution with no unnecessary search. Because the MHD is a minimum over known rather than all possible derivations, it upper-bounds the true minimum. It is a human-legible reference: concise but interpretable as sequential reasoning. Measuring excess above the MHD makes efficiency comparable across problems of different difficulty and gives a fixed target toward which progress can be measured.

We evaluate 15 frontier models from OpenAI and Anthropic on 40 problems from the 2026 AIME and HMMT competitions with repeated sampling, and make four contributions:

- The MHD, a problem-specific benchmark that decomposes each trace into necessary steps and excess, making efficiency comparable across problems and over time.
- Tokens per correct solution fell 8.5× across OpenAI releases (o1→GPT-6-Astra) and 4.8× for Anthropic releases (Opus 4.5→Fable 5.1) while accuracy rose, and within-problem variance fell alongside the mean.
- We estimate that excess above the MHD decays exponentially at 22.3–32.6% per quarter (OpenAI) and 31.7–45.4% (Anthropic), where the lower bound uses only models that cannot have seen the problems. Extrapolating within family, average frontier traces reach within 10% of the MHD by 2027-12 (Anthropic) and 2028-10 (OpenAI).
- Using matched pairs of open-weight models, we suggest that both pre-training scale (gpt-oss-20B vs. 120B) and post-training (GLM 5.2 vs. 5.3) contribute to efficiency gains.

## 2 Related Literature

**The falling cost of inference** The price of a fixed level of benchmark performance has fallen roughly 5–10× per year through hardware, algorithmic, and deployment gains (Gundlach et al., 2025; Ho et al., 2024; Mertens et al., 2026). We study a complementary quantity: the number of tokens to reach a correct answer.

**Overthinking and efficient reasoning** Early reasoning models were prone to overthinking: re-verifying and re-solving even trivial problems (Chen et al., 2024), and RLHF was already known to inflate length (Singhal et al., 2023). Part of the cause of this is a length bias in the standard GRPO objective (Liu et al., 2025; Yu et al., 2025). Optimal chain-of-thought length rises with problem difficulty and falls with model capability (Snell et al., 2024; Wu et al., 2026). A large literature proposes interventions to shorten traces, including length-penalized rewards, inference time budgets, adaptive no-think decisions, and latent-space reasoning (Arora & Zanette, 2025; Aggarwal & Welleck, 2025; Luo et al., 2025; Hou et al., 2025; Han et al., 2025; Muennighoff et al., 2025; Zhang et al., 2025; Hao et al., 2024; Kimi Team, 2025).

**Measuring reasoning token usage** Epoch AI documents that output length has grown over time for both reasoning and non-reasoning models, and that reasoning token usage rises with amount of RL training (Epoch AI, 2025). Their analysis ends in 2025 and captures the rising phase prior to the fall that we document. OckBench jointly scores accuracy and token usage and finds up to 5× differences in tokens between models with similar accuracy (Du et al., 2025). Reasonbench shows that repeated runs of the same model on the same problem vary considerably in cost (Potamitis et al., 2025), and Brown et al. (2024) find that repeated sampling within-problem boosts accuracy. Our per-problem sampling addresses this variation in model responses.

**Chain of thought monitorability** The chain of thought is currently the primary interface for inspecting a reasoning model's process, and Korbak et al. (2025) argue that monitorability is valuable but fragile under training pressure. Baker et al. (2025) find that penalizing undesirable content in the chain of thought teaches models to hide intent rather than abandon it, and Guan et al. (2025) report that monitors perform better on longer chains of thought. We connect these concerns to a measured trend: the efficiency gains we document may carry a cost to oversight (Section 7).

## 3 A Framework for Reasoning Efficiency

**A Search Framework for Model Problem Solving** Our approach is inspired by Newell & Simon (1972), who built a model of human reasoning behavior from think-aloud transcripts, the closest human analog to a chain of thought. They model problem solving as search through a problem space: a set of states, operators that transform one state to the next, and a goal state. Solving a problem is finding a path from the initial state to the goal. Our assumption is not that models solve problems exactly as humans do; rather we utilize the insights from Newell & Simon (1972) to provide definitions of necessary steps and excess search. As illustrated in Figure 1, we treat a reasoning trace as a walk on a directed graph from problem $P$ to solution $S$. Nodes are intermediate states of the derivation, edges are reasoning steps that move from one state to the next. A correct solution is a walk that reaches $S$, the green walk in Figure 1, while a walk into a region from which $S$ is unreachable is a *fatal error*, the red node, and yields an incorrect attempt.

**Experts Search Less** How does search change as a solver becomes more capable? The expertise literature gives us a consistent answer: experts search less, not more. de Groot (1978) found that grandmasters do not search deeper or more extensively than weaker players; they consider better candidate moves from the start. Chase & Simon (1973) traced this to pattern recognition — the experts remembered the pieces in chunks as familiar patterns while novices did not. The same holds outside chess. Expert physicists work forward from the givens towards an answer, while novices proceed backwards through trial and error and take many more steps (Larkin et al., 1980). Machines show the same pattern: in chess, AlphaZero searched orders of magnitude fewer positions per second than Stockfish while achieving superior performance (Silver et al., 2018). Across domains, increased skill results in the number of steps being closer to the minimum number (the green walk in Figure 1), while novices detour and explore.

Applying this to models, a correct trace decomposes into the steps a solution requires and *unnecessary search*: steps that can be removed with the solution intact. Prior work has already recovered this separation through probes in reasoning models (Nikankin et al., 2026). Early reasoning models exhibited the same detours as novices: they backtracked, identifying a step that was wrong and saying "wait, let me reconsider" (DeepSeek-AI, 2025) (the orange node in Figure 1). They also re-verified steps already established and generated redundant solution attempts (Gandhi et al., 2025; Chen et al., 2024) (the purple loop). Further, they were verbose, wrapping each step in explanatory text (Singhal et al., 2023). Removing any of these leaves the green walk in Figure 1 unchanged. This framework therefore predicts that as models become more capable, their traces should shorten towards necessary steps, and they should vary less across repeated attempts of the same problem as they converge to an efficient solution. Section 4 examines both predictions; Section 6 counts verification and backtracking steps in open-weight traces.

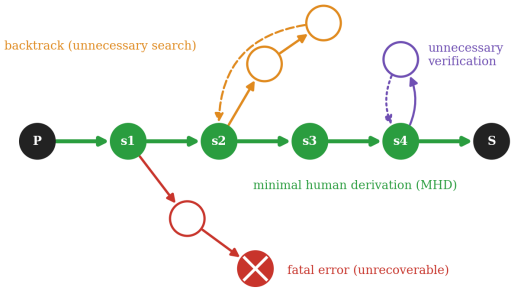



Figure 1: Reasoning as search. Nodes are intermediate problem states; edges are reasoning steps. The green path from the problem $P$ to the solution $S$ is the *minimal human derivation* (MHD): the shortest known sequence of necessary steps. A *fatal error* (red) leaves the solution subspace after $s_1$ and is unrecoverable. A *backtrack* (orange) explores off-path nodes from $s_2$ before returning ("wait, let me reconsider"). An *unnecessary verification* (purple) re-derives $s_4$ without adding a new step.

**The Minimal Human Derivation** The above framework motivates our empirical anchor, the *minimal human derivation* (MHD): for each problem, the shortest known human derivation, tokenized with a model's respective tokenizer. The MHD approximates the green path in Figure 1. It is a post-hoc write-up rather than a record of the solver's process, so it contains only the steps a solution requires. It includes search within the derivation, such as ruling out cases, but not the search for the derivation — any strategies tried and abandoned. Excess reasoning, $E = L - \mathrm{MHD}$, is the tokens a model spends ($L$) above this floor ($\mathrm{MHD}$), and thus measures the search for the derivation and any verbosity. Because the MHD captures each problem's required steps, it decomposes each trace into necessary steps and excess, so that excess is comparable even on problems of different intrinsic difficulty.

The MHD is an attainable floor only when search for a derivation is reducible, that is, when a solver who recognizes a problem's structure can proceed directly. We therefore study competition problems, which are designed to be solved under time pressure by experts who can often find derivations directly. In contrast, problems answered by look-up have no derivation to measure, and on problems where search for a derivation is irreducible, excess would instead converge to an asymptote above the MHD (Section 7). Within our scope, the MHD upper-bounds the true minimum, since published solutions may include exposition a minimal derivation does not need, and a small fraction of frontier traces already fall below it (Section 6). The truly shortest derivation may be neither observable nor desirable, since model outputs should remain comprehensible for human monitors.

## 4 Benchmark and Results

**Problem Set Construction** The expertise literature compares experts and novices on the same problems: what distinguishes them is not whether they solve a problem but how directly they find its solution. We follow the same design, holding the problem set fixed across releases to identify the within-problem trend. The problems must also fall within the scope of Section 3, which competition mathematics does by design, and the short closed-form answers make correctness unambiguous. Our problem set is 40 math problems from the 2026 American Invitational Mathematics Examination (AIME I and II) and the February 2026 Harvard-MIT Mathematics Tournament (HMMT) labeled as medium, hard, or very hard by position within the contest (early AIME, late AIME, and HMMT respectively). Because we study efficiency rather than accuracy, we select problems that models across the timeline can mostly solve, with accuracy ranging from 70% (o1) to 100% (GPT-5.6-sol) among the frontier models. We estimate on correct attempts: a fatal error never reaches the solution (Figure 1), so it has no derivation to compare against the MHD.

The competitions also postdate the training cutoffs of most of our models. The four newest models (Fable 5.1, Opus 5, GPT-5.6-sol, and GPT-6-Astra) have training cutoff dates after these competitions, in GPT-5.6-sol's case two days after the problems were posted, so Figure A7 re-estimates the trend on the eleven models trained before then. Our estimates therefore characterize efficiency on a fixed set of problems solvable across the whole period; efficiency at each release's own frontier is a different question, and one that matters for aggregate cost, since users push models toward their frontier.

**Measuring the Floor** For each problem, we take the human-written solutions posted on the Art of Problem Solving (AoPS) Wiki, which collects independent derivations per problem written by experienced competitors and coaches; for HMMT problems we use HMMT's official solutions. Problems have 1–8 posted solutions (median 2.5; 28 of 40 have at least two), and within a problem the longest is 1.1–4.9$\times$ the shortest. We take the shortest as the minimal human derivation, and use it as a proxy for the shortest known derivation. We tokenize each solution with the tokenizer of the model it is compared against, so that each model's output is compared to its own respective floor.[1]

**Model Evaluations** We measure a trace's length, $L$, as the completion tokens reported by each model provider's API, including both reasoning and final answer tokens, which is what users are billed for. We therefore measure token efficiency, not compute efficiency: a larger model can use fewer tokens but more compute per token. Compute is unobservable for closed models; for the

[1] Anthropic reports changing their tokenizer starting with Opus 4.7 — we use the corresponding tokenizer for each model.

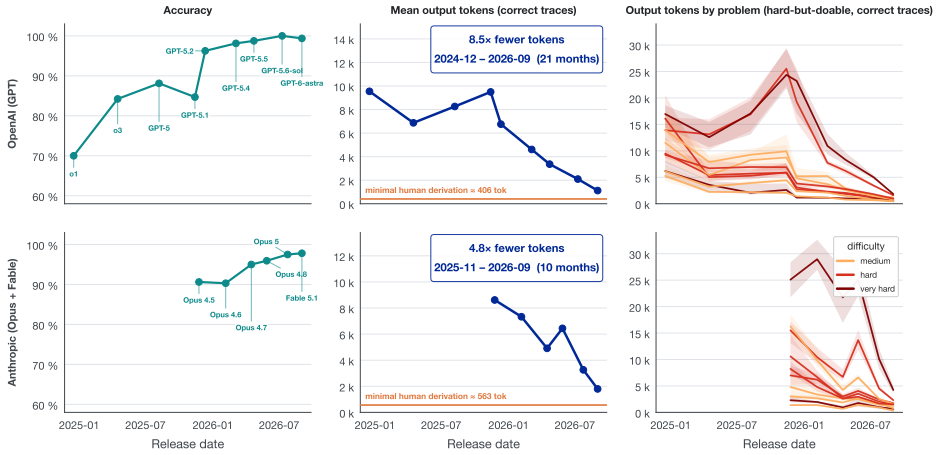



Figure 2: Reasoning efficiency gains over time for frontier model releases from OpenAI and Anthropic, measured on 40 competition problems. **Left:** accuracy on the entire problem set, in teal. **Middle:** Mean output tokens (reasoning and answer) over correct traces, $k = 8$ attempts per problem. The orange horizontal line is the mean length of the minimal human derivation, tokenized per family. **Right:** A "hard-but-doable" subset of 10 problems at $k = 32$ attempts per problem, showing the per-problem median and interquartile range. Colors indicate difficulty, labeled by position within the contest.

open-weight pairs in Section 6, where active parameters are known, we report both. We run every model at "medium" reasoning effort and pin the token cap to 40,000 tokens across flagship model releases[2]. The efficiency trend holds across effort settings (Figure A5). Each problem is attempted $k = 8$ times per model, and $k = 32$ times for the hard-but-doable subset described below.

**Efficiency gains over time** Figure 2 shows the trend. The left panel reports accuracy, showing efficiency gains do not come at the cost of accuracy, which rises from 70% for o1 to 100% for GPT-5.6-sol. Anthropic deprecated its early reasoning models, so its series begins with Opus 4.5 (November 2025). The middle panel reports mean completion tokens over correct attempts, averaging $k = 8$ attempts per problem to smooth run-to-run variation. Because later models solve more problems, their averages include harder problems that earlier models never solved, which biases *against* efficiency gains; Figure A2 repeats the analysis on all attempts.[3] Mean tokens per correct solution fell from ≈9,500 (o1) to ≈1,100 (GPT-6-Astra), an $8.5\times$ reduction over twenty-one months, and from ≈8,600 (Opus 4.5) to ≈1,800 (Fable 5.1), a $4.8\times$ reduction over ten months. The orange line marks the average minimal human derivation, ≈406 tokens under OpenAI's tokenizer and ≈563 under Anthropic's current tokenizer (≈460 before Opus 4.7). Compared with that line, mean traces fell from roughly $23\times$ the average human minimum to $2.8\times$ for OpenAI, and from roughly $19\times$ to $3.2\times$ for Anthropic; per-problem geometric means are shown in Figure 3.

The right panel examines the distribution of attempts within problem over time. We select 10 *hard-but-doable* problems, those with at least an 85% average solve rate across all models, so that every model contributes correct attempts to each problem. We resample each $k = 32$ times (Figure A4 plots the pooled distributions of these problems collapsing toward the floor). We see efficiency manifest in two dimensions: the mean tokens to solve a problem falls, as does the within-problem variance across attempts. This pattern holds at different difficulty levels, shown by the difficulty shading. Falling variance is consistent with our efficiency theories in Section 3: as search improves, there is less variation across attempts as each attempt converges to a minimal path rather than more variable trial and error. We cannot inspect proprietary reasoning traces, but the joint decline in mean and variance is consistent with unnecessary search and verbosity being eliminated.

[2] 1.2% of trials reached the 40k-token cap, concentrated in early high-token models, all graded incorrect, so conditioning on correct solutions removes them and truncation does not affect the token estimates.

[3] On all attempts, compression is $10.9\times$ for OpenAI and $5.6\times$ for Anthropic (Figure A2).

**Variation across releases** Within family, token usage does not fall monotonically. OpenAI's series rises from o3 to GPT-5 and again to GPT-5.1, which is also less accurate (84.7% vs. 88.1%). Part of the GPT-5.1 rise is formatting changes, as answers switch to LaTeX from that release onward.[4] The GPT 5.2 generation is when OpenAI began advertising efficiency in its release notes (OpenAI, 2025a), consistent with earlier models having prioritized capability over efficiency. Claude Opus 4.8 is less efficient than its predecessor Opus 4.7, which may reflect an especially verbose model release.

## 5 Forecasting Efficiency

To estimate how fast excess reasoning has declined and when it approaches the floor, let $i$ index model within a family (ordered by release), $j$ index problem, $t = 1, \ldots, k$ index repeated attempts. Let $L_{ijt}$ denote the completion-token count of attempt $t$ by model $i$ on problem $j$, and let $\mathrm{MHD}_j$ denote the token length of the minimal human derivation for problem $j$. Excess reasoning is $E_{ijt} = L_{ijt} - \mathrm{MHD}_j$. Throughout, we condition on correct attempts, so that $E_{ijt}$ measures the excess cost of a solution rather than a failure.

We estimate

$$\log\big(L_{ijt} - \mathrm{MHD}_j\big) \;=\; \alpha_j \;+\; \beta\, m_i \;+\; \varepsilon_{ijt}, \tag{1}$$

where $m_i$ is model $i$'s release date in months and $\alpha_j$ is a problem fixed effect. Subtracting $\mathrm{MHD}_j$ makes length decay toward a problem's own floor rather than an impossible zero-token solution. We take logs because efficiency gains are plausibly proportional: each generation removes a fraction of the remaining excess, implying exponential decay toward the floor. The log also stabilizes variance across problems of very different lengths. Problem fixed effects absorb intrinsic differences in per-problem difficulty. We use a monthly time index because releases are unevenly spaced (roughly quarterly, sometimes shorter), and we estimate within model family, as providers standardize reasoning settings and tokenizers differently. $\beta$ is the monthly proportional rate at which the gap to the floor closes; we report it as an equivalent quarterly decline, $1 - e^{3\beta}$ and the half-life $-\ln 2/\beta$. Standard errors are wild cluster bootstrapped by model.

Since the logarithm requires $E_{ijt} > 0$, we exclude the 0.7% (32 of 4,435) of correct traces that fall at or below the floor which are concentrated in the recent models. 10 come from GPT-6-Astra (9 below the floor, 2.8% of its correct traces, and one exactly at it), 8 from Opus 4.7 (2.6% of correct traces), and 14 from Fable 5.1 (4.5% of correct traces). We analyze below-floor traces further in Section 6.

Excess over the floor decays at $32.6\%$ per quarter for GPT models, a half-life of $5.3$ months. For Anthropic, we find a decay rate of $45.4\%$ per quarter and a half-life of $3.4$ months. Extrapolating the trend puts model efficiency within $25\%$ of the minimal human derivation by 2028-03 for OpenAI and 2027-08 for Anthropic, and within $10\%$ by 2028-10 and 2027-12 respectively. These are forecasts from a short series and carry considerable uncertainty; the Anthropic estimate in particular is significant only at the 10% level (bootstrap $p = 0.068$). We do not make claims about whether the Anthropic trend is faster than OpenAI; the confidence intervals overlap.

Two caveats apply to the fitted rate. First, easy gains may have come first, so decay may slow. Second, restricting to models with cutoffs before the competitions gives 22.3% per quarter for OpenAI and 31.7% for Anthropic (pre-cutoff columns of Table 1; Figure A7). We report both as bounds, since the gap is some combination of acceleration and contamination we cannot separate; the lower bound is conservative because early GPT releases did not target efficiency, visible in the flatter early segment of Figure 2. Dropping any single model leaves the decline intact, though the newest model in each family is the most influential (Table A1).

[4] We do not strip LaTeX, as users are billed for it. Re-estimating on reasoning tokens alone in Figure A3 and Table A2 addresses any formatting changes that may have occurred across model releases. GPT-5.1's token usage is flat from GPT-5 when examining reasoning tokens alone.

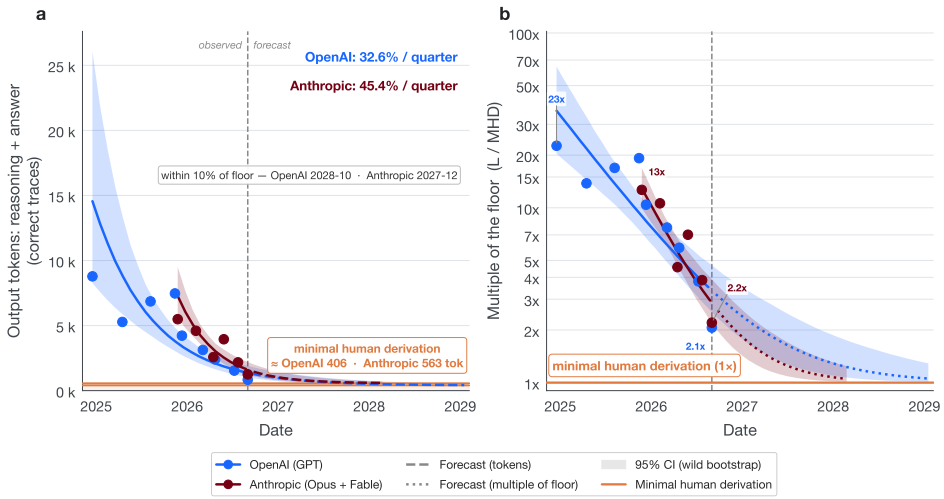



Figure 3: Reasoning length declines toward the human derivation floor for OpenAI (o1→GPT-6-Astra) and Anthropic (Opus 4.5→Fable 5.1). **Panel a:** Dots are per-model geometric means over correct traces, in output tokens and **Panel b:** as a multiple of each problem's MHD. The first and last model of each family are labeled. Solid curves are the fit to Eq. 1, dashed curves extrapolate it, and shaded bands are 95% wild-cluster-bootstrap intervals over the 6–9 models per family. Forecast dates are the mean, across problems, of the month at which each problem's fitted excess falls within 10% of its own MHD.

| | OpenAI (GPT) | | Anthropic (Opus + Fable) | |
|---|---|---|---|---|
| | All models | Pre-cutoff | All models | Pre-cutoff |
| Release month | −0.132 | −0.084 | −0.202 | −0.127 |
| | (0.025) | (0.014) | (0.035) | (0.045) |
| Problem fixed effects | Yes (40) | Yes (40) | Yes (40) | Yes (40) |
| Observations | 2,610 | 1,982 | 1,793 | 1,182 |
| Model clusters | 9 | 7 | 6 | 4 |
| Bootstrap-$t$ $p$ | 0.025 | 0.055 | 0.068 | 0.085 |
| Quarterly reduction | 32.6% | 22.3% | 45.4% | 31.7% |
| Half-life (months) | 5.3 | 8.3 | 3.4 | 5.5 |

Table 1: Excess reasoning declines with release date. OLS regressions of $\log(L_{ijt} - \mathrm{MHD}_j)$ on release month with problem fixed effects, on correct attempts, for all models and for the pre-cutoff subsample whose training cutoffs predate the competitions. Wild-cluster-bootstrap standard errors (clustered by model) in parentheses; bootstrap-$t$ $p$-values from 9,999 replications with Webb weights. Quarterly reduction $= 1 - e^{3\hat{\beta}}$; half-life $= -\ln 2/\hat{\beta}$.

## 6 Sources of Gains and the Frontier

**Scale and post-training** The set of reasoning models is still too small for large-scale time-series analysis to estimate contributions of pre-training and post-training on efficiency. Instead, we examine matched pairs of open-weight models that vary one lever at a time. For scale, we compare gpt-oss-20B with gpt-oss-120B released in August 2025, two mixture-of-experts models (21B and 117B total parameters, 3.6B and 5.1B active per token) trained with the same recipe according to their model card, although OpenAI does not expose details (OpenAI, 2025b). For post-training, we examine GLM 5.2 (released June 2026) to GLM 5.3 (August 2026); both have 744B total, 40B active parameters. Z.ai states that GLM 5.3 and 5.2 share a base model (Z.ai, 2026), with 5.3 having gone through much more extensive RL post-training. Both pairs are run at matched effort settings

within pair (medium for gpt-oss, high for GLM, which offers no medium setting) so trace lengths are comparable within family but not across the two pairs.

Both scale and post-training contribute to efficiency gains in Figure A1. Accuracy rises modestly (68% to 74% and 83% to 86%), and mean tokens per correct solution falls $1.7\times$ in both ($7{,}842 \rightarrow 4{,}693$ tokens for scale and $14{,}241 \rightarrow 8{,}539$ for post-training). The within-problem distributions differ, however. Scale compresses the whole distribution, narrowing the interquartile range on all 10 hard-but-doable problems, and the median moves down with it. Post-training lowers the mean without tightening the distribution, the coefficient of variation rises from $0.35$ to $0.53$, so run-to-run variability becomes proportionally larger even as the mean shortens.

**Inside the traces** We count two behaviors that correspond to unnecessary search in our framework: verification (the purple loop in Figure 1), using the prompt of Gandhi et al. (2025), and backtracking (the orange node), using a prompt of our own,[5] each applied by an LLM judge to whole traces. Table 2 reports both per correct trace and per 10k reasoning tokens. With $5.6\times$ more total parameters ($1.4\times$ more active per token), verifications per trace fall 49% and backtracks 41% in a trace 43% shorter, so their density per 10k tokens is roughly unchanged. The larger model takes about half as many detours to reach the same answer, though each token costs roughly $1.4\times$ the compute, so compute per correct solution falls by only about a fifth. Because the GLM models share an architecture, their compute per correct solution falls in proportion to tokens (46%).

With post-training, mean reasoning tokens fall 46% (the median 72%)[6] while verifications per trace fall only 24% and backtracks per trace rise 45%, so the density of both rises. GLM 5.3 searches at least as much as 5.2 but writes far less around it, consistent with our finding that the variance does not fall with the mean. GLM 5.3 also attempts to recall an answer before deriving on 30% of traces, but the recalled answer is never correct and never replaces the derivation (Appendix A).

| | Scale | | Post-training | |
|---|---|---|---|---|
| | gpt-oss-20B | gpt-oss-120B | GLM 5.2 | GLM 5.3 |
| Verification, per trace | 9.07 | 4.59 | 12.24 | 9.28 |
| Verification, per 10k reasoning tokens | 13.39 | 11.83 | 9.33 | 12.99 |
| Backtracking, per trace | 2.35 | 1.39 | 1.57 | 2.28 |
| Backtracking, per 10k reasoning tokens | 3.46 | 3.57 | 1.20 | 3.19 |
| Mean reasoning tokens | 6,775 | 3,879 | 13,112 | 7,146 |
| Median reasoning tokens | 4,509 | 2,850 | 10,074 | 2,862 |
| Correct traces judged | 214 | 236 | 264 | 267 |

Table 2: Verification and backtracking markers per correct trace and per 10k reasoning tokens, from the LLM judges. Token figures are reasoning tokens only. All rows use the same judged correct traces: the 15 traces on which the judge exceeded its output limit are excluded from both behaviors (Appendix A).

**Reaching the Minimal Human Derivation** The pace of reasoning model efficiency shows most clearly at the frontier (Figure A8). Nearly two years ago, the best reasoning model used roughly $23\times$ the shortest human derivation. Today, 14% of GPT-6-Astra's correct solutions are shorter than the *average* published expert solution, and 2.8% fall below the minimum; the comparable numbers for Fable 5.1 are 16% and 4.5%. The two models get there differently. Astra emits a thinking block on every sub-floor trace,[7] while Fable 5.1 emits no thinking tokens on about a quarter of correct

[5]The backtracking prompt of Gandhi et al. (2025) counts expressions of doubt ("wait", "hmm") as backtracks; in our framework doubt that ends by confirming a step is verification, so we count a backtrack only when a trace raises a candidate approach and abandons it. Both judges were validated against hand counts on a sample.

[6]Figure A1 shows reasoning and answer, hence the differing numbers.

[7]Astra's release was accompanied by reports of a recurrent-depth architecture (Efrati et al., 2026), which would allow computation in latent space rather than in tokens (Hao et al., 2024). If so, its token counts would understate its reasoning; we cannot verify this from the API.

attempts and derives the solution directly in the answer block. All but one of Fable 5.1's sub-floor traces are of this kind.

## 7 DISCUSSION

Nearly two years after reasoning models were introduced, tokens per correct solution have fallen 8.5× across OpenAI releases and 4.8× across Anthropic releases, and excess reasoning above the minimal human derivation decays at 22.3–32.6% and 31.7–45.4% per quarter, respectively. The decline holds across difficulty levels and within problems: later models use fewer tokens and vary less across repeated attempts of the same problem. The newest models approach, and occasionally fall below, the length of the shortest human derivation.

The pattern is consistent with the expertise literature: while novices explore, backtrack, and re-verify, experts recognize the structure of a problem and proceed directly, taking a similar path on repeated encounters (de Groot, 1978; Chase & Simon, 1973; Larkin et al., 1980). Falling means and variance fit that picture, but token counts alone cannot establish it: they do not separate pattern recognition from recall on the newest models, or fewer detours from terser prose. In open-weight traces, both occur: scale removes detours, while post-training mainly removes the prose around them (Section 6); decomposing proprietary traces would show which dominates at the frontier.

Because inference is billed per token, tokens per problem solved are a direct cost to users. Two forces have compounded to make a given capability cheaper: the price per token has fallen roughly 5–10× per year through hardware and algorithmic gains (Gundlach et al., 2025), and, as we show, tokens needed to solve a given problem have fallen roughly 3.4–6.6× per year. Erol et al. (2025) define the cost-of-pass, the expected cost of a correct answer, and document its decline. Our work shows one channel through which the decline occurs: tokens per correct answer decaying toward a floor.

Finally, shorter traces may come at a cost for oversight. Chain of thought is the primary interface for inspecting a model's reasoning, and monitors are shown to perform better when chains of thought are longer (Guan et al., 2025). Pressure to shorten reasoning traces, whether from length penalties or market pressure on token consumption, creates incentives to reason less visibly (Korbak et al., 2025; Baker et al., 2025). Our data show that Fable 5.1 already emits zero thinking tokens on a quarter of correct attempts, writing the derivation in the answer block instead, and both Astra and Fable 5.1 produce traces shorter than the minimal human derivation on 3–5% of correct attempts. Chain of thought monitoring is currently among the best tools for catching misaligned behaviors such as scheming and reward hacking. The efficiency gains we document mark real progress in capabilities and user cost, yet they may be a double-edged sword.

**Limitations** Reasoning models are two years old, so each family contributes 6–9 releases and the wild-cluster bootstrap intervals are correspondingly wide; adding problems would not narrow them, since the trend is identified across releases. We study competition mathematics on a fixed set of problems solvable by every model in the series; gains may differ in domains without verifiable answers, and efficiency at each release's own frontier of difficulty is a separate quantity that may not fall (Epoch AI, 2025).

Our measure is tokens, not compute or search. Tokens are what users pay for and what monitors see, but compute per token differs across models and is unobservable for closed ones. The MHD upper-bounds the true minimal derivation, since published solutions may include exposition a derivation does not need. Our scope assumes search for a derivation is reducible on these problems; if it is not for some, excess converges to an asymptote above the MHD and our arrival dates are optimistic. Eq. 1 imposes the MHD as the asymptote; rates move by about 3 percentage points and arrival dates by up to 5 months across alternative floors (Table A2), and the dates should be read as conditional on that choice.

Finally, comparisons across releases rest on provider settings we cannot hold fixed. We evaluate at "medium" effort and estimate within family, but providers may re-tune what medium means; the trend appears at all three effort settings (Figure A5). Four of fifteen models have cutoffs after the competitions and may have seen the problems or their published solutions; the pre-cutoff estimates in Table 1 bound that effect.

### AI USE STATEMENT

In this work, we used generative AI tools for producing figures via Claude Code and writing code to pass to the OpenAI/Anthropic APIs for running the problem set, which was verified by the authors. We also used AI for assistance in finding related literature which was verified by the authors. The framework and concepts were developed by the authors, AI was used to polish writing and refine delivery.

### ETHICS STATEMENT

We have no ethical considerations to state.

### REPRODUCIBILITY STATEMENT

All problems are drawn from the publicly available 2026 AIME I, AIME II, and February 2026 HMMT contests; the minimal human derivations are the shortest solutions posted on the Art of Problem Solving Wiki and HMMT website as of June 2026, and Section 4 describes their selection and per-model tokenization. Section 4 and footnotes specify the evaluation protocol: model identifiers, reasoning-effort setting, the 40,000-token cap, the number of attempts per problem ($k = 8$ for the full problem set, $k = 32$ for the hard-but-doable subset), and grading by mathematical equivalence to the answer key. The regression specification, sample restrictions, and wild-cluster-bootstrap procedure are given in Section 5 and Table 1; alternative floor specifications and the pre-cutoff subsample are in Appendix A. The LLM-judge prompts used for the trace analysis in Section 6 are in Appendix A (Figures A9–A10), together with the judge models. Anonymous code is uploaded via supplementary material; the full repo will be made public upon acceptance. Because proprietary reasoning traces are not returned by either provider, the reported token counts are the completion-token counts from each provider's API at the time of collection; we note that provider-side changes may affect exact replication.

## REFERENCES

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

Daman Arora and Andrea Zanette. Training language models to reason efficiently. In *Advances in Neural Information Processing Systems*, 2025. arXiv:2502.04463.

Bowen Baker, Joost Huizinga, Leo Gao, Zehao Dou, Melody Y. Guan, Aleksander Madry, Wojciech Zaremba, Jakub Pachocki, and David Farhi. Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. *arXiv preprint arXiv:2503.11926*, 2025.

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

William G. Chase and Herbert A. Simon. Perception in chess. *Cognitive Psychology*, 4(1):55–81, 1973.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT think that much for 2+3=? on the overthinking of o1-like LLMs. *arXiv preprint arXiv:2412.21187*, 2024.

Michelene T. H. Chi, Paul J. Feltovich, and Robert Glaser. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5(2):121–152, 1981.

Adriaan D. de Groot. *Thought and Choice in Chess*. Mouton, The Hague, 2nd edition, 1978. Original Dutch edition 1946; first English edition 1965.

Alex de Vries. The growing energy footprint of artificial intelligence. *Joule*, 7(10):2191–2194, 2023.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. Also published in *Nature*, 2025.

Zheng Du, Hao Kang, Song Han, Tushar Krishna, and Ligeng Zhu. OckBench: Measuring the efficiency of LLM reasoning. *arXiv preprint arXiv:2511.05722*, 2025.

Amir Efrati, Stephanie Palazzolo, and Rocket Drew. OpenAI technique in 'Astra' model sparks security concerns. *The Information*, September 2026. URL https://www.theinformation.com/articles/secret-technique-behind-openais-astra-model-sparks-security-concerns.

Epoch AI. LLM responses to benchmark questions are getting longer over time. https://epoch.ai/data-insights/output-length, April 2025.

Mehmet Hamza Erol, Batu El, Mirac Suzgun, Mert Yuksekgonul, and James Zou. Cost-of-pass: An economic framework for evaluating language models. *arXiv preprint arXiv:2504.13359*, 2025.

Fortune. After blowing through AI budget in a matter of months, Uber CTO says tokenmaxxing era is over. *Fortune*, August 2026. URL https://fortune.com/2026/08/07/uber-ai-spending-tokenmaxxing-is-over-cto/.

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective STaRs. *arXiv preprint arXiv:2503.01307*, 2025.

Melody Y. Guan, Tony Wang, Micah Carroll, et al. Evaluating chain-of-thought monitorability. OpenAI, https://openai.com/index/evaluating-chain-of-thought-monitorability/, December 2025.

Hans Gundlach, Jayson Lynch, Matthias Mertens, and Neil Thompson. The price of progress: Price-performance and the future of AI. *arXiv preprint arXiv:2511.23455*, 2025.

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*, 2025.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

Anson Ho, Tamay Besiroglu, Ege Erdil, David Owen, Robi Rahman, Zifan Carl Guo, David Atkinson, Neil Thompson, and Jaime Sevilla. Algorithmic progress in language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Bairu Hou, Yang Zhang, Jiabao Ji, Yujian Liu, Kaizhi Qian, Jacob Andreas, and Shiyu Chang. ThinkPrune: Pruning long chain-of-thought of LLMs via reinforcement learning. *arXiv preprint arXiv:2504.01296*, 2025.

Kimi Team. Kimi k1.5: Scaling reinforcement learning with LLMs. *arXiv preprint arXiv:2501.12599*, 2025.

Tomek Korbak, Mikita Balesni, Elizabeth Barnes, et al. Chain of thought monitorability: A new and fragile opportunity for AI safety. *arXiv preprint arXiv:2507.11473*, 2025.

Jill Larkin, John McDermott, Dorothea P. Simon, and Herbert A. Simon. Expert and novice performance in solving physics problems. *Science*, 208(4450):1335–1342, 1980.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. In *Conference on Language Modeling (COLM)*, 2025. arXiv:2503.20783.

Sasha Luccioni, Yacine Jernite, and Emma Strubell. Power hungry processing: Watts driving the cost of AI deployment? In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2024. arXiv:2311.16863.

Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao, and Dacheng Tao. O1-Pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. *arXiv preprint arXiv:2501.12570*, 2025.

Matthias Mertens, Natalia Fischl-Lanzoni, and Neil Thompson. Is there "secret sauce" in large language model development?, 2026. URL https://arxiv.org/abs/2602.07238.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

Nature News. OpenAI claims huge maths breakthrough on a famed 'millennium problem'. *Nature*, September 2026. News article, d41586-026-02842-5.

Allen Newell and Herbert A. Simon. *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, NJ, 1972.

Yaniv Nikankin, Martin Tutek, Tomer Ashuach, Jonathan S. Rosenfeld, and Yonatan Belinkov. Reasoning models know what's important, and encode it in their activations. In *Conference on Language Modeling (COLM)*, 2026. arXiv:2604.18307.

OpenAI. Learning to reason with LLMs. https://openai.com/index/learning-to-reason-with-llms/, September 2024.

OpenAI. Introducing GPT-5.2. https://openai.com/index/introducing-gpt-5-2/, December 2025a.

OpenAI. gpt-oss-120b & gpt-oss-20b model card. arXiv preprint arXiv:2508.10925, August 2025b.

OpenAI. On the Navier–Stokes millennium prize problem. https://openai.com/index/navier-stokes-solution/, September 2026.

Nearchos Potamitis, Lars Klein, and Akhil Arora. ReasonBENCH: Benchmarking the (in)stability of LLM reasoning. *arXiv preprint arXiv:2512.07795*, 2025.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. A long way to go: Investigating length correlations in RLHF. *arXiv preprint arXiv:2310.03716*, 2023.

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.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. In *International Conference on Learning Representations*, 2026. arXiv:2502.07266.

Qiying Yu, Zheng Zhang, Ruofei Zhu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Z.ai. GLM-5.3. https://z.ai/blog/glm-5.3, August 2026. Released 14 August 2026. "It uses the same base model as GLM-5.2 — every gain comes from post-training.".

Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. AdaptThink: Reasoning models can learn when to think. *arXiv preprint arXiv:2505.13417*, 2025.

# A APPENDIX

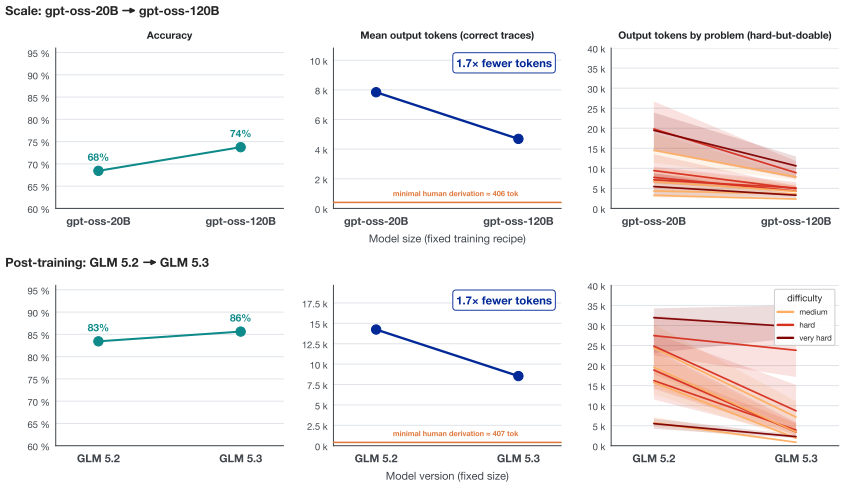



Figure A1: Where do efficiency gains come from: scale or post-training? **Top:** scale, gpt-oss-20B→120B at a fixed training recipe. **Bottom:** post-training, GLM-5.2→5.3 on a shared base model. Columns follow Figure 2: accuracy, mean output tokens over correct traces with the MHD floor ($k = 8$, 40 problems), and per-problem median and IQR on the hard-but-doable subset ($k = 32$), shaded by difficulty. Vertical scales differ between rows and are not comparable: gpt-oss was run at medium effort and GLM at high.

**GLM 5.3 Recall Attempts** Before any computation, GLM 5.3 attempts to recall an answer on 30% of its traces and states a remembered answer on 18%; none of these remembered answers are correct, and it cites contests from 1984–2025, never the 2026 contests the problems come from. In every case it then derives the answer, and the derivation, not the recalled answer, is what it reports. Recall therefore does not explain GLM 5.3's efficiency: the recalled answer never replaces the derivation. The other three models attempt recall on 1% or fewer of traces and never state a remembered answer. It appears to have learned to attempt recall before derivation through its extensive post-training.

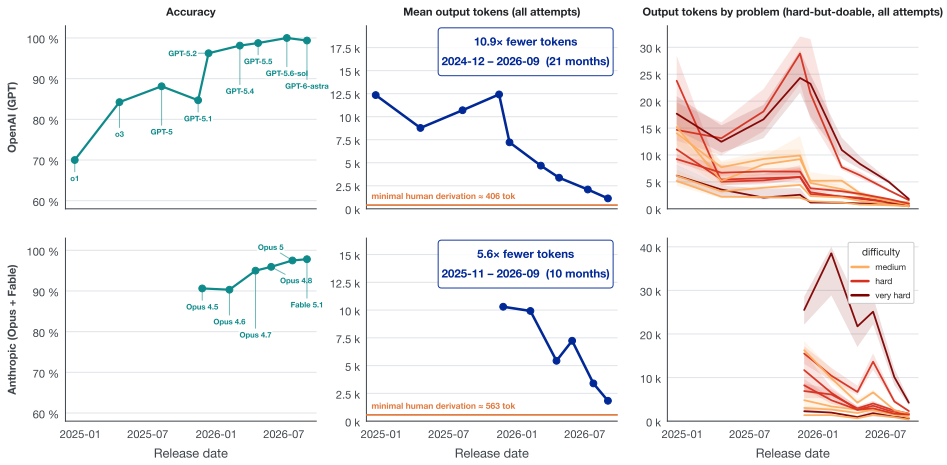



Figure A2: A replication of Figure 2 but with all traces to account for selection on success. Conditioning on correct traces is conservative: repeating the analysis on all attempts including failures raises the compression from $8.5\times$ to $10.9\times$ for OpenAI and $4.8\times$ to $5.6\times$ for Anthropic.

| Model left out | $\hat{\beta}$ (SE) | Quarterly reduction | Within 10% of floor | Observations |
|---|---|---|---|---|
| *OpenAI (GPT)* (9 models; 8 per refit) | | | | |
| None (all models) | $-0.132$ $(0.025)$ | 32.6% | 2028-10 | 2,610 |
| o1 | $-0.152$ $(0.035)$ | 36.5% | 2028-06 | 2,386 |
| o3 | $-0.145$ $(0.031)$ | 35.3% | 2028-07 | 2,343 |
| GPT-5 | $-0.129$ $(0.025)$ | 32.1% | 2028-11 | 2,328 |
| GPT-5.1 | $-0.129$ $(0.021)$ | 32.1% | 2028-10 | 2,339 |
| GPT-5.2 | $-0.131$ $(0.023)$ | 32.4% | 2028-10 | 2,302 |
| GPT-5.4 | $-0.134$ $(0.024)$ | 33.0% | 2028-09 | 2,296 |
| GPT-5.5 | $-0.134$ $(0.025)$ | 33.2% | 2028-09 | 2,294 |
| GPT-5.6-sol | $-0.130$ $(0.030)$ | 32.3% | 2028-11 | 2,290 |
| GPT-6-Astra | $-0.104$ $(0.017)$ | 26.8% | 2029-09 | 2,302 |
| *Anthropic (Opus + Fable)* (6 models; 5 per refit) | | | | |
| None (all models) | $-0.202$ $(0.035)$ | 45.4% | 2027-12 | 1,793 |
| Opus 4.5 | $-0.237$ $(0.042)$ | 50.9% | 2027-09 | 1,503 |
| Opus 4.6 | $-0.195$ $(0.033)$ | 44.4% | 2028-01 | 1,504 |
| Opus 4.7 | $-0.203$ $(0.038)$ | 45.6% | 2027-12 | 1,497 |
| Opus 4.8 | $-0.213$ $(0.028)$ | 47.2% | 2027-11 | 1,486 |
| Opus 5 | $-0.215$ $(0.040)$ | 47.5% | 2027-11 | 1,481 |
| Fable 5.1 | $-0.150$ $(0.021)$ | 36.3% | 2028-08 | 1,494 |

Table A1: Leave-one-model-out refits of the decay regression (1). Each row drops one model and refits on the same 40 problems, dependent variable and wild-cluster bootstrap; "None" is the headline fit. The slope stays negative in every refit. The most influential models are each family's endpoints: dropping the newest model (GPT-6-Astra, Fable 5.1) lowers the quarterly reduction to 26.8% and 36.3%, and dropping the oldest (o1, Opus 4.5) raises it to 36.5% and 50.9%. Removing a model removes a cluster (8 per OpenAI refit, 5 per Anthropic refit); the question is whether any one model moves the slope, not whether each refit is separately significant.

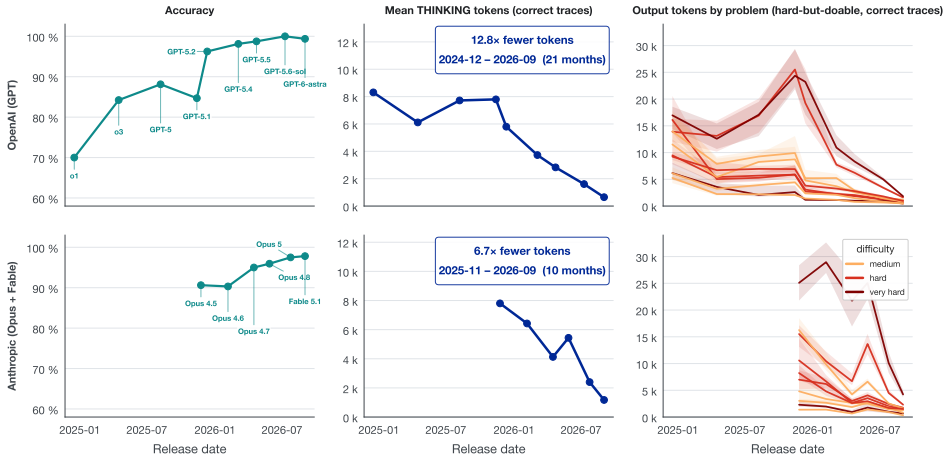



Figure A3: A replication of Figure 2 but with only reasoning tokens to account for any formatting changes across model releases, such as switching to LaTeX. GPT-5.1 average token usage is flat from GPT-5 when looking at only reasoning tokens. We recalculate the decay rate with only reasoning tokens in Table A2; decay rate is similar although this is partially mechanical and only illustrative as it removes the floor.

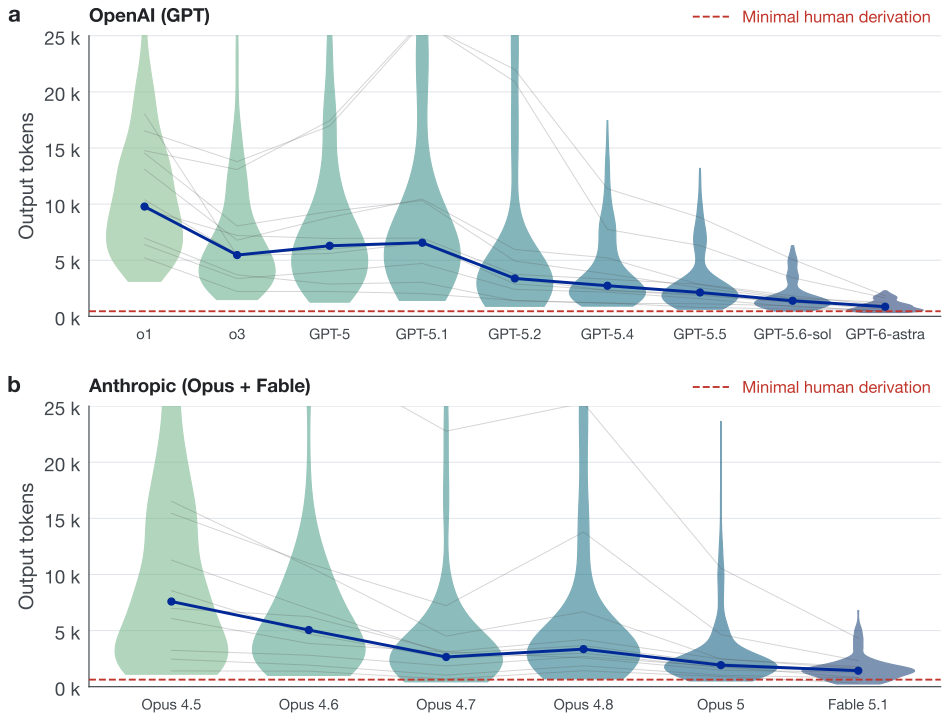



Figure A4: Hard-but-doable problem distributions. Tokens over time collapse towards the minimal human derivation both in terms of the mean tokens outputted per problem and the variance of attempts within problem.

**Robustness Checks on the MHD** Table A2 reports alternative floor specifications and a reasoning tokens only fit.

**LLM Judge for Verification and Backtracking Markers** We use a lighter judge for verification (Gemini-2.5-Flash), a simpler detection task following the prompt of Gandhi et al. (2025) and a

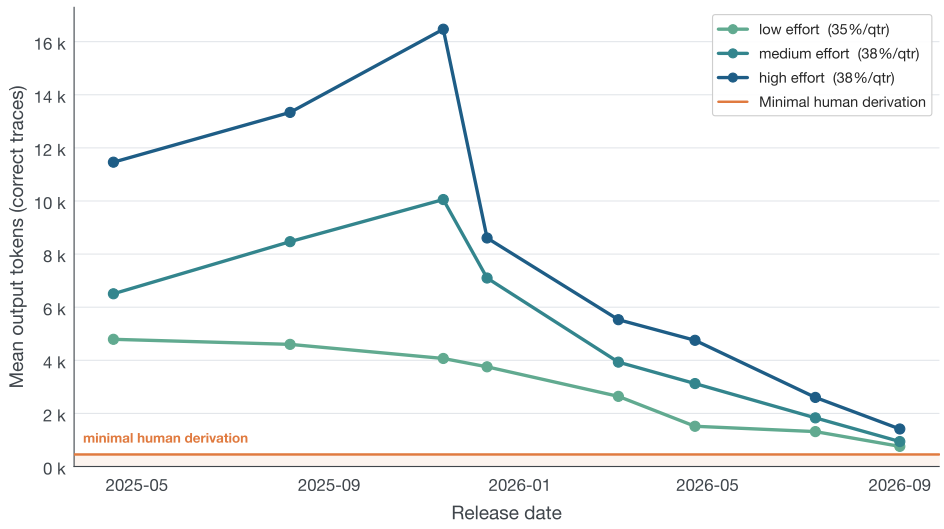



Figure A5: Decay rate is robust across reasoning effort settings, we use the hard-but-doable sample of problems rather than our full problem set at $k = 8$ attempts for the OpenAI series (note this is a smaller sample of problems than our main text, thus the decay rate is slightly different), and find similar decay rates across low, medium and high effort settings.

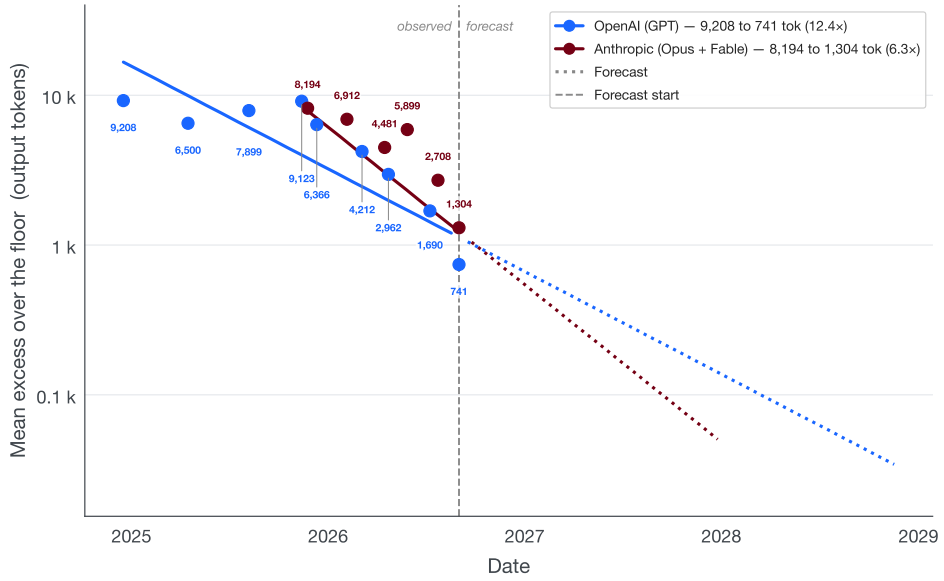



Figure A6: Mean excess over the floor, $L_{ijt} - \mathrm{MHD}_j$, in output tokens (arithmetic mean over correct traces, log scale). Figure 3 reports geometric means and multiples of the floor; this figure shows the same decline in raw tokens above the floor. Solid lines are the fit to Eq. 1 and dotted lines their extrapolation.

stronger judge for backtracking (Gemini-3.1-Pro), which requires tracking an approach across the trace until it is abandoned; both were validated against hand counts on a sample. Both judges were run on every correct trace. On 15 traces, the verification judge exceeded its 65,535-token output limit before returning a count; the backtracking judge scored every trace. We drop these 15 traces from both behaviors so that verification and backtracking are measured on the same set. The dropped traces are among the longest, and computing backtracking on all correct traces instead changes the between-model backtracking ratios in Table 2 by at most 0.03.

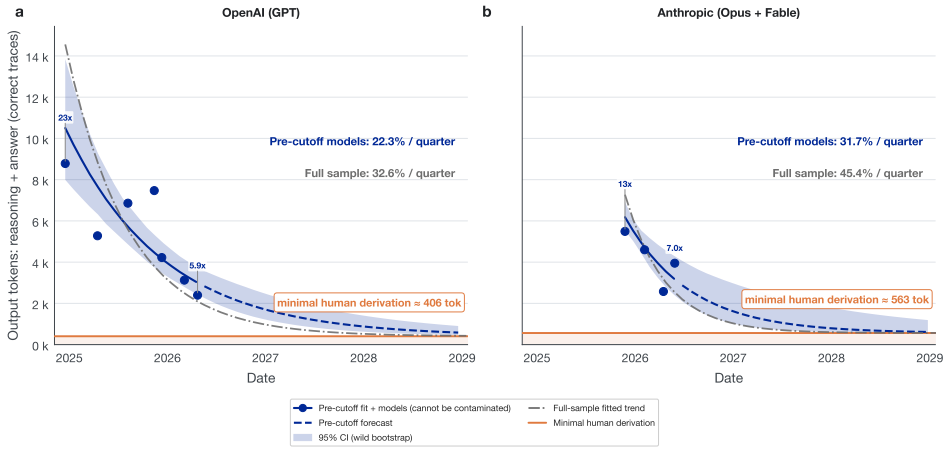



Figure A7: To address problem set contamination concerns, we refit our model with only models with training cutoff dates prior to our problem set release date in February 2026. The decay rate decreases, from 32.6% to 22.3% for OpenAI and from 45.4% to 31.7% for Anthropic. The sample comprises seven OpenAI and four Anthropic models. The lower rates partly reflect that pre-2026 OpenAI releases did not prioritize efficiency; GPT-5.1's slight rise is in part due to a switch to LaTeX-formatted answers (Figure A3).

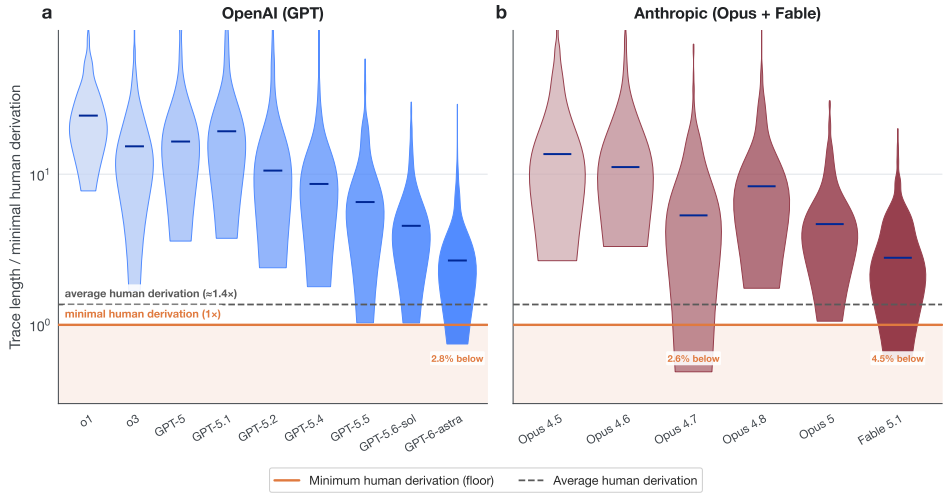



Figure A8: Trace lengths relative to the minimal human derivation, OpenAI (left) and Anthropic (right). Each violin pools the problem set at $k = 8$ repeated samples per problem. Distributions are over correct attempts, the solid line at 1 marks the MHD and the dashed line marks the average human derivation length. Astra falls below the minimum on 2.8% of responses and below the average human derivation on 14%. Fable 5.1 has 16% of answers below the average human derivation and 4.5% below minimal.

| Family | Floor $\mathrm{MHD}_j$ | $\overline{\mathrm{MHD}}_j$ (tok) | $\hat{\beta}$ (SE) | Quarterly reduction | Half-life (months) | Within 10% of floor |
|---|---|---|---|---|---|---|
| OpenAI (GPT) | shortest | 406 | $-0.132$ (0.025) | 32.6% | 5.3 | 2028-10 |
| | median | 526 | $-0.140$ (0.026) | 34.4% | 4.9 | 2028-06 |
| | mean | 541 | $-0.142$ (0.026) | 34.7% | 4.9 | 2028-05 |
| | thinking only | — | $-0.134$ (0.023) | 33.1% | 5.2 | — |
| Anthropic (Opus + Fable) | shortest | 563 | $-0.202$ (0.035) | 45.4% | 3.4 | 2027-12 |
| | median | 731 | $-0.220$ (0.036) | 48.2% | 3.2 | 2027-09 |
| | mean | 748 | $-0.217$ (0.035) | 47.9% | 3.2 | 2027-09 |
| | thinking only | — | $-0.182$ (0.016) | 42.1% | 3.8 | — |

Table A2: Alternative specifications to Eq. 1, estimated per family on correct attempts over the 40 competition problems with problem fixed effects, wild cluster SE over the models in parenthesis. "Shortest" is the specification in the main text. The mean and median rows replace $\mathrm{MHD}_j$ with the mean and median published solution, estimated rates move by about three percentage points and the extrapolated 10% of the floor date moves by up to five months, reflecting the shifting lower bound. The thinking-only rows fit $\log L$ on reasoning tokens alone, excluding answer tokens, to control for formatting changes across releases (Figure A3). They subtract no floor, so no arrival date is reported and the rate is not directly comparable to the floored rows and reported only for robustness. The thinking-only rates of $33.1\%$ (OpenAI) and $42.1\%$ (Anthropic) indicate that reasoning tokens decline at a similar rate as total output.

```
# Task Description
You will be provided with the reasoning trace of a language model solving a competition mathematics
    problem.
Evaluate whether the text contains any verification steps. We want to mark instances where the writer
    explicitly checks their own work, such as by comparing the result to a known value or by checking the
    result of a calculation.

Verification steps in mathematics might look like:
- "Let's check our answer by substituting x = 3 back into the original equation."
- "To verify this is correct, I'll differentiate the antiderivative and confirm it matches the original
    function."
- "Let's test our formula with a simple case: when n = 1, we get f(1) = 2, which matches our expected
    result."
- "To ensure this solution is valid, I'll check if it satisfies all the given constraints."

If you find any verification steps, please count them and provide the count between the tags <count>
    </count>. If the text does not contain any verification steps, please provide a count of 0 as
    <count>0</count>.

# Task Format
Format your response in markdown as follows:

## Thoughts
[Brief description describing what behavior was noticed and where answer verification may have occurred]

## Does verification?
[yes/no]

## Number of answer verification steps
<count> [1/2/...] </count>

# Task to evaluate for Verification
{response}

# Response
```

Figure A9: Prompt for the LLM Judge to count verification markers in the text, modified slightly from Gandhi et al. (2025) to fit the competition math setting. We manually verify on a sample to ensure judge accuracy. Judge model is Gemini-2.5-Flash at temperature 0.

```
# Task Description
You will be provided with the reasoning trace of a language model solving a competition mathematics
    problem. Each line of the trace is prefixed with its line number (L1:, L2:, ...).

Mark each place where the writer tries an approach and gives it up to do something different. If the
    writer comes back to the same approach, or continues it after a pause, that is one entry, not several.

Do not mark:
- Checking a step that was already completed (that is verification).
- Ruling out a case that the solution has to rule out.

# Task Format
For each place, in order of appearance:

<instance>
approach: [short name of the approach that was given up]
adopt_line: [line number where the attempt starts]
adopt_quote: "[5-15 words copied exactly from that line]"
abandon_line: [line number where it is given up]
abandon_quote: "[5-15 words copied exactly from that line]"
</instance>

After the last instance (or immediately, if there are none), give the number of places:
<count> [0/1/2/...] </count>

# Task to evaluate
{response}

# Response
```

Figure A10: Prompt for the LLM Judge to count backtracking markers in the text, modified from Gandhi et al. (2025) to fit the competition math setting. We count a candidate approach raised and abandoned, and exclude doubt resolved by confirmation. Validated against hand counts on a sample. Judge is Gemini-3.1-Pro at temperature 0.