HIDDEN PATTERNS IN CHAIN-OF-THOUGHT REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Chain-of-thought (CoT) prompting is a de-facto standard technique to elicit reasoning-like responses from large language models (LLMs), allowing them to spell out individual steps before giving a final answer. While the resemblance to human-like reasoning is undeniable, the driving forces underpinning the success of CoT reasoning still remain largely unclear. In this work, we perform an in-depth analysis of CoT traces originating from competition-level mathematics questions, with the aim of better understanding how, and which parts of CoT actually contribute to the final answer. To this end, we introduce the notion of a *potential*, quantifying how much a given part of CoT increases the likelihood of a correct completion. Upon examination of reasoning traces through the lens of the potential, we identify surprising patterns including (1) its often strong non-monotonicity (due to reasoning *tangents*), (2) very sharp but sometimes tough to interpret spikes (reasoning *insights* and *jumps*) as well as (3) at times *lucky guesses*, where the model arrives at the correct answer without providing any relevant justifications before. While some of the behaviours of the potential are readily interpretable and align with human intuition (such as insights and tangents), others remain difficult to understand from a human perspective. To further quantify the reliance of LLMs on reasoning *insights*, we investigate the notion of CoT *transferability*, where we measure the potential of a weaker model under the partial CoT from another, stronger model. Indeed aligning with our previous results, we find that as little as 20% of partial CoT can "unlock" the performance of the weaker model on problems that were previously unsolvable for it, highlighting that a large part of the mechanics underpinning CoT are transferable.

1 INTRODUCTION

Chain-of-thought (CoT) reasoning (Wei et al., 2023) has led to several breakthroughs in domains ranging from mathematics (Cobbe et al., 2021; Gao et al., 2023; Luo et al., 2025; DeepSeek-AI et al., 2025) to coding (Chen et al., 2021; Austin et al., 2021; Li et al., 2022; Lozhkov et al., 2024; Rozière et al., 2024), enabling modern language models to now win gold medals at mathematical olympiads (Luong et al., 2025). The underlying idea of CoT is very simple and intuitive: let the model reason through the given problem and explain its steps before giving a final answer. This approach offers two main advantages: (1) Generating additional tokens means more computation available to the model, allowing it to implement more complex routines. (2) CoT enables the model to decompose complex problems into more manageable sub-tasks, akin to human reasoning.

The success of chain-of-thought reasoning is undeniable, yet the precise mechanisms driving it remain poorly understood. A very tempting explanation, due to their (by design) strong resemblance to human reasoning, is that LLMs similarly benefit from spelling out bigger computations more slowly, using techniques such as backtracking and verification to explore several avenues before finally arriving at the best answer (Zhou et al., 2023; Shinn et al., 2023; Madaan et al., 2023b; Press et al., 2023). Other works however suggest that the content of CoTs might not always reflect the actual solving strategy of the model, for instance Lanham et al. (2023); Chen et al. (2025b) show that the model's explanations to addition task do not line up with the underlying computation performed internally. This result seems to rather suggest that CoTs primarily act as computational mechanisms, letting the model execute more complicated algorithms or heuristics "under the hood" while at the same time mimicking human reasoning.

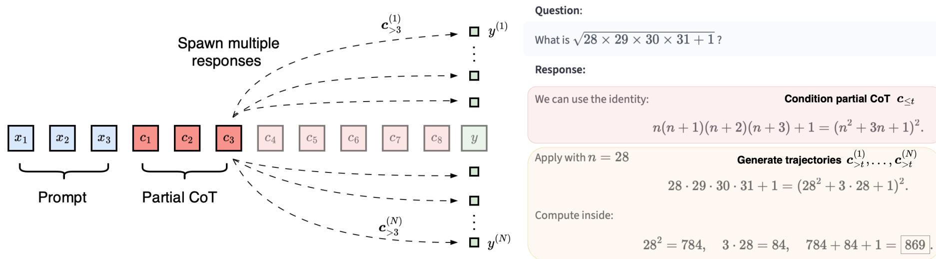

Figure 1: **Left:** Illustration of the calculation of the potential. **Right:** An example prompt and partial CoT, which in this case should intuitively raise the probability of success (i.e. the potential) significantly once discovered or provided by another model.

These perspectives motivate a closer look at how CoT actually contributes in practice. We therefore closely examine reasoning traces produced by several models with a focus on competition-level mathematics questions from AIME-2024, AIME-2025 (MAA, 2025) and MATH-500 (Hendrycks et al., 2021), GPQA-Diamond (Rein et al., 2023) for more general reasoning, as well as coding problems from HumanEval (Chen et al., 2021). In the main text we focus on AIME, as its difficult questions present an ideal arena to study properties of reasoning chains, especially as modern models still produce highly variable CoTs for the same question, *sometimes* leading to the correct solution, but often failing to do so. In this paper, we aim to understand what, or which parts of a CoT make it successful or wrong? To gain insights into this question, we introduce the notion of the *potential*, defined as the probability of success of the model when sampling conditioned on a given partial chain of the CoT (see Eq. 1 for a precise definition). As the potential initially starts out low (models can only sometimes arrive at the right answer), we can use it to monitor precisely which tokens (or collection thereof) increase or decrease it, equipping us with a tool to understand what parts of CoT unlock a previously difficult problem. We observe that similarly to humans, LLMs often exhibit reasoning *insights*, i.e., strong increases of the potential due to the completion of a conceptually difficult step (see e.g., Fig. 1, 5, 9, 11, 12, 13, 14 or 15). Not all spikes in the potential are easily interpretable however; we find that performance can significantly increase through seemingly trivial steps, coined reasoning *jumps* (see e.g. Fig. 5 or Fig. 6) Surprisingly, we observe that the potential is far from monotonic, i.e. not every token contributes effectively towards the final answer but rather long durations of no progress or even sharp drops can occur. The latter are often due to reasoning *tangents*, i.e. approaches which initially look promising but ultimately lead to dead ends or even wrong answers, (see e.g. Fig. 9, 12, 13, 14 or 15).

To further study the usage of reasoning insights in language models, we investigate the degree of *transferability* of CoT between different models. We focus on providing a weaker model with the (partial) CoT from a stronger one, with the motivation that if models indeed struggle with conceptual understanding of the problem, their reasoning might be unblocked when being provided correct sub-steps. Indeed, difficult mathematical questions often involve solving several steps of non-uniform difficulty, with some problems even becoming mostly trivial for humans once a specific insight is obtained or provided. An illustrative example of such a question is shown in Fig. 1, taken from AIME-1989 (MAA, 2025). While the question might look intimidating to many math students at first sight, the problem becomes easily solvable when presented with the insight that $n(n+1)(n+2)(n+3)+1=(n^2+3n+1)^2$. In other words, human reasoning is often able to transfer if the gap is not too large. For LLMs, we find similar results; problems that were previously unsolved by the weaker model, gradually become solvable as more and more CoT is provided, even as little as 20% of CoT leads to a significant improvement in performance. We observe such transferability even between very different model classes, e.g. Qwen3-0.6B's accuracy significantly improves when provided with partial CoT of GPT-OSS-20B. This suggests that language models can profit from insights provided by stronger models, suggesting that some CoTs work in a model-agnostic way.

2 Related Work

Chain-of-Thought reasoning has been very influential in recent years, with every modern language model now being trained to give reasoning-like responses. This characteristic has been strongly

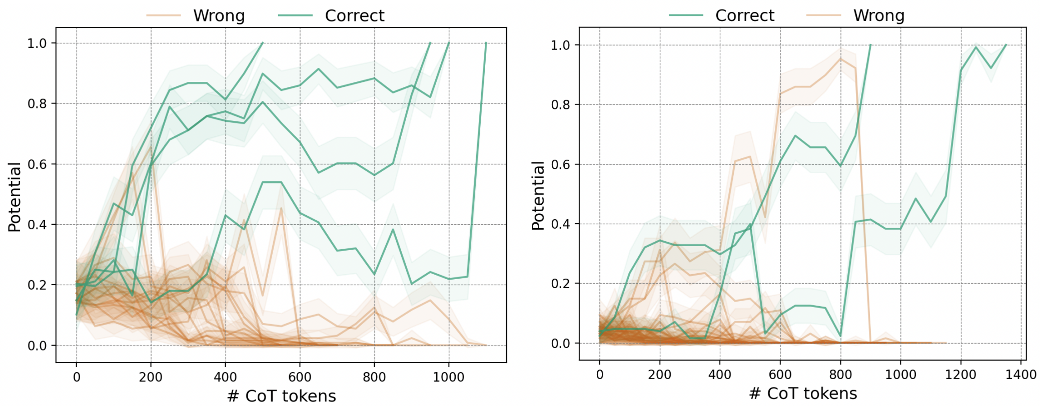

Figure 2: **Potential curves.** Potential of correct and wrong CoTs for Qwen2.5-7B on AIME-2024, Question 5 and 11. Strongly non-monotonic behaviour for both correct and incorrect CoTs.

exacerbated by the emergence of reasoning models such as o1 (OpenAI et al., 2024) and R1 (DeepSeek-AI et al., 2025), further encouraging longer responses by training with reinforcement learning with verifiable rewards. Such models now regularly require generating $128k$ tokens for difficult mathematics questions before returning a final answer. The still human-like nature of these reasoning chains has inspired a surge of works with the aim of interpreting and understanding how these long sequences of tokens actually contribute to the final answer. A line of work has investigated how models react when their CoT is manipulated through insertion of mistakes (Wang et al., 2023) or changes in symbols (Madaan et al., 2023a; Madaan & Yazdanbakhsh, 2022), finding them to be surprisingly robust. Other works have investigated several attribution strategies to identify important parts in CoT (Golovneva et al., 2023; Berchansky et al., 2024; Wu et al., 2023). Opposite types of findings have also been made; Lyu et al. (2023); Lanham et al. (2023); Madsen et al. (2024) have observed that CoT does not always reflect the underlying computation of the model, making it thus difficult to pin-point helpful steps in the first place. Other works go a step further and argue that CoT reasoning should not be compared to human reasoning (Kambhampati et al., 2025; Stechly et al., 2025; Bhambri et al., 2025) or that they outright imitate reasoning without actually performing any (Shojaee et al., 2025). Finally, the line of works most similar to ours also studies conditional generation from partial CoTs; Bigelow et al. (2025) investigate so-called “fork tokens” in the context of neural text generation. Bogdan et al. (2025) also explore the notion of conditional generation to find “thought anchors”, parts of CoT that help the model arrive at correct answers. While their focus is on more abstract reasoning concepts such as backtracking and self-verification, we focus on task-relevant insights and also explore the failure modes of CoT reasoning. Finally, Amani et al. (2025) also explore the notion of completing partial CoTs, incorporating the idea in reinforcement learning for better reward signal. Our latter definition of the potential shares strong resemblance to the value function in actor-critique models (Konda & Tsitsiklis, 1999; Sutton & Barto, 2018), similarly measuring the quality of a given state but in a Monte-Carlo fashion.

3 Potential of CoT

Setup. Let $\mathcal{V}$ denote the vocabulary. Assume we have a tokenized input prompt $\boldsymbol{x} \in \mathcal{V}^D$ (e.g., encoding a math question) and a ground truth answer $y^* \in \mathcal{V}$ (for simplicity represented by a single token) encoding the expected response (e.g. “513”). Let $\texttt{LM}_\theta$ represent a language model with parameters $\boldsymbol{\theta}$, mapping a sequence of tokens $\boldsymbol{x}$ to the logits of size $|\mathcal{V}|$. When answering to a prompt, models now generate $T \in \mathbb{N}$ intermediate *chain-of-thought* tokens $\boldsymbol{c} \in \mathcal{V}^T$ auto-regressively before arriving at a final answer y. I.e. given prompt $\boldsymbol{x}$, we generate $c_t \sim \texttt{LM}_\theta(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x})$ auto-regressively and only then sample the answer, $y \sim \texttt{LM}_\theta(\bullet|\boldsymbol{c}, \boldsymbol{x})$. Generations involving such intermediate tokens have been observed to outperform models trained (or prompted) to directly provide an answer in a variety of settings (Wei et al., 2023). We will often abuse notation slightly by letting $(y, \boldsymbol{c}_{\geq t}) \sim \texttt{LM}_\theta(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x})$ denote the (sequential) autoregressive generation, conditional on $(\boldsymbol{c}_{<t}, \boldsymbol{x})$. Typical decoding strategies in language models leverage this stochastic generation and

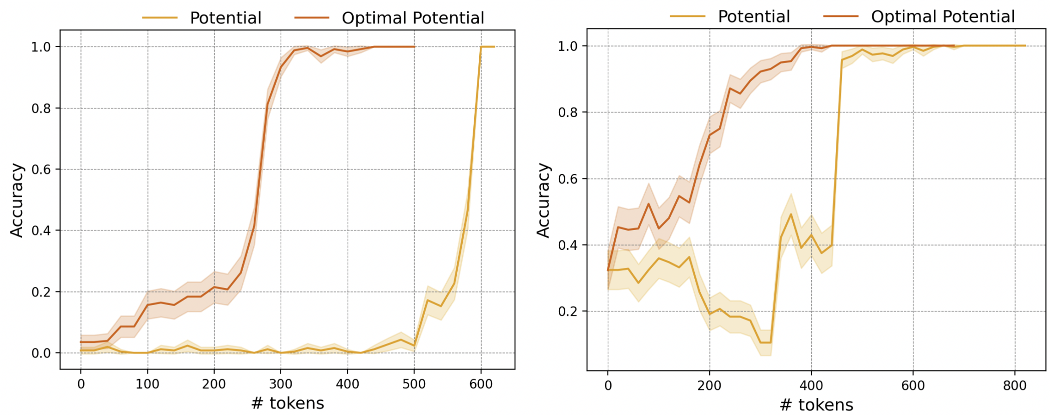

Figure 3: The potential of optimal and standard CoT for AIME-2025-I, question 1 and 5. While standard CoT eventually arrives at the right answer, optimal CoT does so in a more robust way.

it is hence interesting to consider $K \in \mathbb{N}$ such generations by varying the random seeds, either unconditionally or starting from a partial CoT $\boldsymbol{c}_{<t}$,

$$\left(y^{(k)}, \boldsymbol{c}_{\geq t}^{(k)}\right) \overset{i.i.d.}{\sim} \text{LM}_{\theta}(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x}) \quad \text{for } k = 1, \ldots, K$$

where we obtain most likely distinct CoT completions $\boldsymbol{c}_{\geq t}^{(k)}$ and final answers $y^{(k)}$.

Potential. Given a prompt $\boldsymbol{x}$ and an associated reasoning process $\boldsymbol{c}$ with final answer y, it is natural to ask which sub-steps in $\boldsymbol{c}$ contributed most to the overall result. Let us define the *potential* of a chunk of CoT $\boldsymbol{c}_{<t}$ on the prompt $\boldsymbol{x}$ as the probability of correct generation conditioned on $\boldsymbol{c}_{<t}$,

$$\text{pot}(\boldsymbol{c}_{<t}; \boldsymbol{x}) := \mathbb{P}_{(\boldsymbol{c}_{\geq t}, y) \sim \text{LM}_{\theta}(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x})} \left(y = y^*\right) \tag{1}$$

Intuitively, if a chunk of CoT is useful or encompasses a step that the model tends to struggle with, conditioning on it should subsequently lead to a higher potential. In mathematical terms, if conditioning on a shorter prefix $\boldsymbol{c}_{<s}$ for $s < t$ has a lower potential compared to $\boldsymbol{c}_{<t}$ i.e. $\text{pot}(\boldsymbol{c}_{<s}; \boldsymbol{x}) < \text{pot}(\boldsymbol{c}_{<t}; \boldsymbol{x})$, this implies that the CoT chunk $\boldsymbol{c}_{s<t}$ "made progress" towards the final solution. On the other hand, if the potential remains similar, $\text{pot}(\boldsymbol{c}_{<s}; \boldsymbol{x}) \approx \text{pot}(\boldsymbol{c}_{<t})$, then the chunk of CoT $\boldsymbol{c}_{s<t}$ did not solve a step that is difficult to the model, as it can reliably reproduce it under sampling. This does not necessarily imply that such steps can be skipped as they could entail necessary computations such as a long multiplication, which the model can reliably do but also *needs* to do. Finally, we can have situations where the potential decreases, with CoTs actively worsening the state of the model. On average however, we can show mathematically that the potential improves monotonically over all correct CoTs:

Proposition 1. *Conditional on the event that the full CoT $\boldsymbol{c}_{1:T}$ yields the correct final answer y^*, it holds for every $t \leq T$ that*

$$\mathbb{E}\left[\text{pot}(\boldsymbol{c}_{<t}; \boldsymbol{x})\right] \leq \mathbb{E}\left[\text{pot}(\boldsymbol{c}_{<t+1}; \boldsymbol{x})\right].$$

We invite the reader to check the proof in Appendix A.2. Hence on average, every token c_t should push the potential higher, encouraging the model to converge towards the correct solution, reflecting the intuition that chain-of-thought performs *evidence accumulation*. Calculating the potential exactly is unfortunately intractable, so in practice we use the following estimator instead,

$$\text{pot}_N(\boldsymbol{c}_{<t}; \boldsymbol{x}) := \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{y^{(n)} = y^*\}} \quad \text{where } \left(y^{(n)}, \boldsymbol{c}_{\geq t}^{(n)}\right) \sim \text{LM}_{\theta}(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x})$$

Sampling a higher number of trajectories N will provide a better approximation to the true potential. We observe that setting $N = 128$ gives very reliable estimations of the potential and use it throughout this work. We provide more experimental details regarding the computational complexity in Appendix A.5.

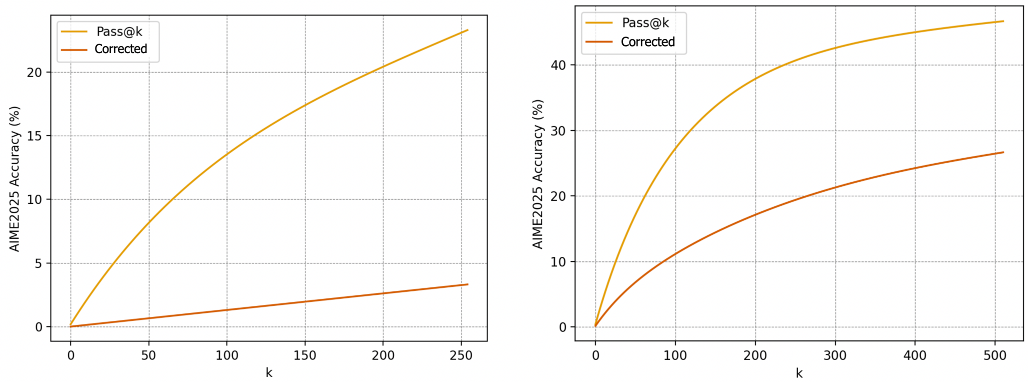

Figure 4: **Inflated** $\text{pass}@k$. We show $\text{pass}@k$ accuracies and the corresponding corrected values for Qwen2.5-1.5B (left) and Qwen2.5-7B (right).

4 SHAPE OF POTENTIAL CURVES

We now empirically study the potential $\text{pot}(\boldsymbol{c}_{<t}; \boldsymbol{x})$ as a function of the CoT chunk length t. When conditioning on CoTs $\boldsymbol{c}$ that lead to the correct answer $y = y^*$, based on Prop. 1, we expect the potential to be a smooth and monotonic function in t, with every chunk of CoT $\boldsymbol{c}_{s<t}$ positively contributing to the overall solution. We will mainly focus on difficult competition-level mathematics questions, where the potential $\text{pot}(\boldsymbol{c}_{<0}; \boldsymbol{x})$ corresponding to the "empty" CoT $\boldsymbol{c}_{<0}$ is strictly between 0 and 1, i.e. the model only sometimes produces the correct answer when prompted from scratch. If the model is always correct, the potential does not offer any insight into the CoT; all steps are equally easy to the model. In contrast, if performance starts significantly lower, we can precisely pinpoint where a successful CoT overcame hurdles that stopped most other attempts. We calculate the potential curves for a variety of models, including both the non-thinking types of models Qwen2.5 (sizes 1.5B and 7B), (Qwen et al., 2025) and Llama-3.1 (sizes 8B and 70B) (Grattafiori et al., 2024), as well as the reasoning models Qwen3 (sizes 0.6B and 32B) (Yang et al., 2025). We display a variety of potential curves (both for correct and wrong trajectories) in Fig. 2 for two samples taken from AIME-2024. Surprisingly, typical chain-of-thought exhibits quite erratic potentials, with certain sections of CoT actively worsening the probability of success, going against the theoretical result in Prop. 1. We will examine the characteristics of potential curves qualitatively in close detail in Sec. 5. We quantify the following properties of potential curves often exhibited across AIME-2024: (1) Very sharp increases in the potential in a small token window, we will later refer to these occurrences as reasoning *insights* and *jumps*. (2) Very sharp drops in the potential, we coin this behaviour reasoning *tangents* or *flaws*. (3) Extremely late increases in the potential, which previously remained flat and close 0. We will show qualitatively in Sec. 5 that such CoTs are very often associated with *guessing*, i.e. the model produces a correct answer without relying on its previously generated reasoning and at times even admits to do so.

MODEL	REASONING	INSIGHTS ↑	TANGENTS ↓	LATE SPIKE	MONOTONICITY
QWEN2.5-1.5B	✗	40%	5%	20%	45%
QWEN2.5-7B	✗	62%	9.5%	14%	42%
LLAMA3.1-8B	✗	46%	33%	6%	15%
LLAMA3.1-70B	✗	37%	40%	5%	17%
QWEN3-0.6B	✓	55%	41%	10%	15%
QWEN3-32B	✓	36%	18%	0%	36%

Table 1: Behaviours of potential for several reasoning and non-reasoning models on AIME-2024.

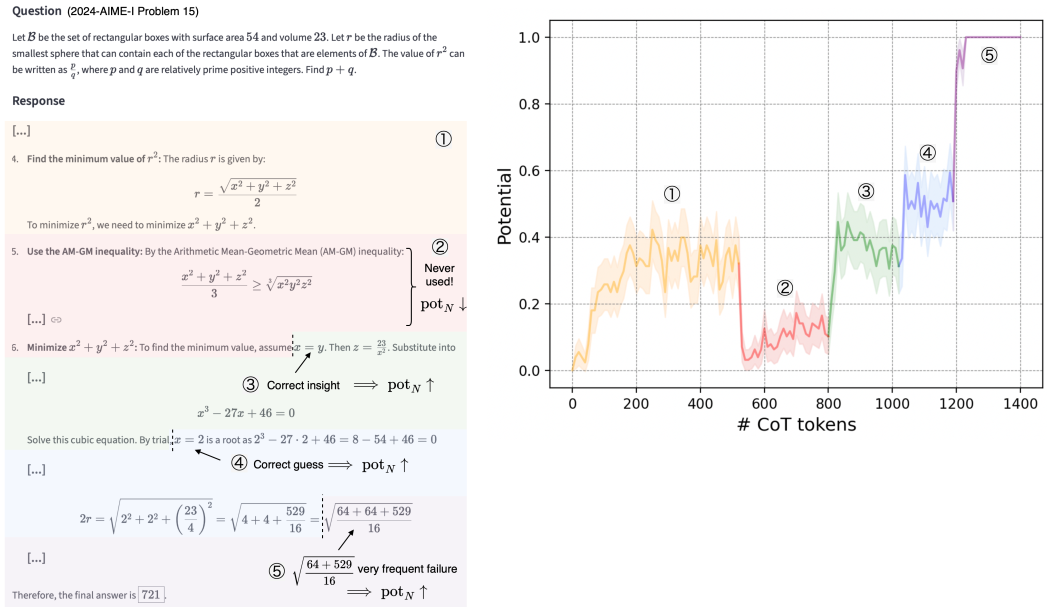

Figure 5: **Reasoning tangents and insights.** Qwen2.5-7B's potential $\mathrm{pot}_{256}(\bullet; \boldsymbol{x})$ behaving strongly non-monotonic. The reasoning *tangent* ② hurts the potential, while the reasoning *insights* ③ (observing the symmetry $x = y$ of the problem) and ④ (finding the root of the cubic equation) push the potential back on track. Finally, the model performs a reasoning *jump* ⑤ (for some non-obvious reason, this particular calculation is difficult for the model).

Quantifying the shape. In the following, we will derive some quantitative summary statistics corresponding to the observations we made based on the plots in Fig. 2. We calculate the potentials for 128 responses per sample on AIME-2024 (total of 30×128 samples) and filter out responses that led to wrong answers. We further only consider samples that are difficult enough for the given model to not reach perfect accuracy without any partial CoT. We then derive four summary statistics that aim to describe the properties introduced above, we detail their precise definitions in Appendix A.7. We show analogous results on MATH-500 in Table 2 in the Appendix.

We display the results in Table 1. Our initial observations are substantiated; only half of the CoTs exhibit monotonicity, with reasoning models tending to produce even more erratic potentials. Non-reasoning models seem to exhibit more late spikes, which aligns with our qualitative observations later in Sec. 5 that such models tend to produce correct answers often through guessing on very difficult problems. Model size also seems to surpress this behaviour more, which is expected since larger models generally tend to perform better. Reasoning tangents occur more often for reasoning models, aligning well with the observation in the literature that such models have the tendency to *overthink* (Chen et al., 2025a), i.e. they discard the discovered, correct answer and explore alternative but flawed approaches. This also partially explains their less monotonic potential. All models exhibit a high amount of reasoning insights, suggesting that most of the difficulty is concentrated in a few key steps instead of being uniformly spread out, more akin to human reasoning.

Amount of guessing. We now focus on the situations where the potential spikes very late in the reasoning process, which we connect almost uniquely with the model *guessing* answers correctly. This has a strong effect on the $\mathrm{pass}@k$ metric used to assess model capabilities, which we show is strongly impacted in the case of Qwen2.5-1.5B and 7B when used in conjunction with weak verification such as final answer verification in mathematical benchmarks. For a dataset consisting of P queries $\{\boldsymbol{x}_i\}_{i=1}^P$ with corresponding answers $\{y_i^*\}_{i=1}^P$, we sample k responses $y_i^{(j)}$ per question from the model and measure if the correct answer is at least once among this set, i.e.

$$\mathrm{pass}@k = \frac{1}{P}\sum_{i=1}^{P} \mathbb{1}_{\left\{y^* \in \{y_i^{(1)}, \ldots, y_i^{(k)}\}\right\}}$$

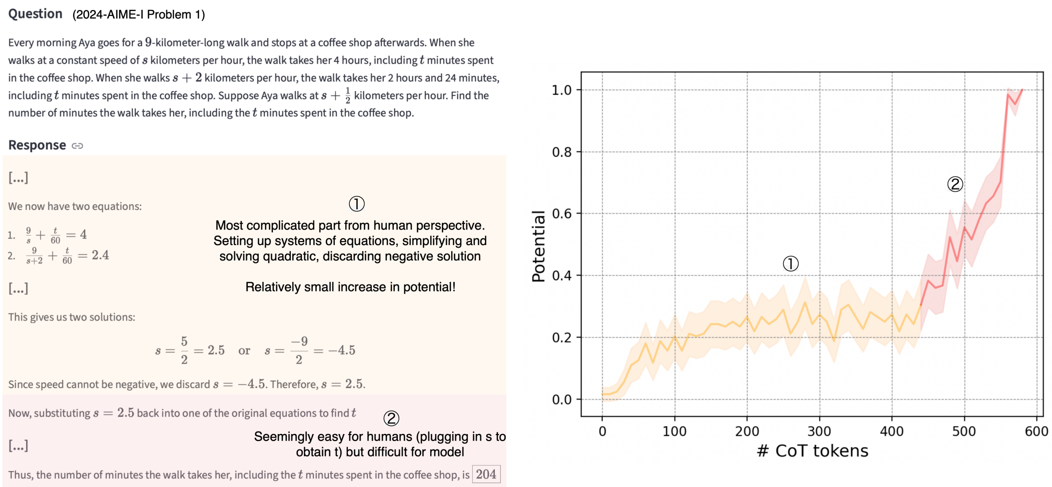

Figure 6: **Unaligned difficulty.** Qwen2.5-1.5B solves most difficult parts in ① but only small increase in potential. Seemingly easier part ② of just obtaining t given s and adding the two turns out to be significantly more difficult.

The idea behind this metric is to measure whether the model does possess the ability to sometimes achieve the right answer, albeit not reliably. Especially for large k, this metric could fall victim to lucky guesses as (1) it only takes one correct answer to obtain the full score and (2) the reasoning process is usually not being assessed in the case of mathematics benchmarks. Indeed, in Fig. 4 we show that the $\text{pass@}k$ scores can be very inflated by flagging samples with the *late spike* statistic, in this case on AIME-2025.

Optimizing the potential. Given our observation that CoT does not naturally follow a monotonic curve, with many tokens even worsening performance, the following question emerges:

Can we search the space of CoT $\mathbf{c}$ such that every sub-CoT $\mathbf{c}_{s<t}$ contributes?

One way to try and maximize the potential of every chunk of CoT is to set a chunk size $C \in \mathbb{N}$ and randomly explore candidate chunks, calculate their potential and keep the highest scoring chunk. In this manner, we can construct a CoT that increases the potential at least gradually if the model admits such reasoning, ideally avoiding issues such as reasoning tangents. We summarize the recipe in Algorithm 1 and the incurred computational complexity in Appendix A.6 more formally. We indeed find that models admit such optimized CoT, we display some associated potential curves in contrast with regular CoT in Fig. 3. We can indeed see that the optimized CoT displays strong monotonicity with most tokens contributing to the potential. This is in stark contrast with the standard CoT, which either does not increase the potential for a long token horizon (left side of the figure), or even actively worsens it (right side). We examine such CoTs more qualitatively in Appendix A.3.

5 A Closer Look at Chain-of-Thought Reasoning

We now perform a qualitative analysis of various chain-of-thought reasonings on competition-level mathematics. Due to the verbosity of reasoning models such as Qwen3, we limit this section to the Qwen2.5 series, whose CoT is more concise and thus more amenable to direct interpretation. The only exception is Fig. 7, where we display parts of a trace from Qwen3-0.6B. Our goal is to precisely align the potential curve with the underlying reasoning produced by the model, and as a consequence obtain an understanding of the types of tokens that drive or hinder the progress. For space reasons we defer from displaying the full CoT but instead show only the sections crucial to the potential. We refer the interested reader to Appendix A.3 for additional qualitative examples.

We display the first sample obtained from Qwen2.5-7B in Fig. 5, a potential curve that exhibits strong non-monotonicity as we have often encountered (see Fig. 2). We dissect the reasoning into five segments according to the potential. Segment ① steadily makes progress towards the solution

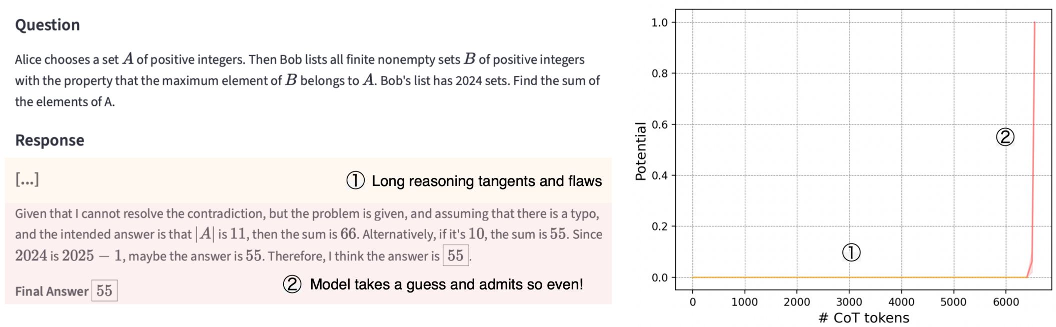

Figure 7: **Reasoning tangents and guessing.** Qwen3-0.6B goes on a long reasoning tangent in ① that does not increase the potential over a long token horizon. Finally it outputs a final answer in ②, itself admitting that the guess is not backed by the reasoning prior but seems likely to the model.

by correctly expressing the radius as a function of the sides of the box and formulating the optimization problem. In segment ② the model goes on a reasoning *tangent*, a step that is not necessarily wrong but happens to not work out for the particular problem (*AM-GM inequality* gives a non-tight lower bound for the minimum). The model manages to ignore this step in this particular trajectory, but on average suffers from this distraction, leading to a sharp drop in the potential. In segment ③ and ④, the model correctly recognizes the symmetry of the problem as well as discovers the root of the cubic equation, with both *insights* consequently boosting the accuracy akin to human reasoning. Finally, in segment ⑤ we observe the final spike in the potential, stemming from a simple arithmetics step that the model tends to get wrong. While the previous spikes were readily interpretable, the last one seems more unintuitive, given that the model manages to very reliably perform the arguably harder arithmetics steps just before. We coin this a reasoning *jump*, a very sharp increase in the potential that largely seems due to a very model-specific issue.

Such misalignment in perceived difficulty of sub-steps is often present in CoT, in Fig. 6 we display another reasoning trace of Qwen2.5-7B along with the associated potential which exhibits this surprising characteristic. Segment ① here does the conceptual heavy-lifting; it correctly deduces the associated system of equations in two variables, simplifies and obtains the solution for the first variable s. The completion of these seemingly involved steps is only rewarded with a small increase in potential, as opposed to humans, the model does not struggle here. Instead, the more difficult steps contained in segment ② consist of now obtaining the second variable t, which only involves plugging the value for s into the previously derived equation. Compared to the previous segment, finishing the problem starting from the end of ① would be a significantly simpler task for humans.

Another surprising insight we obtained is that models can be very capable of guessing solutions to such problems. In Fig. 7 (and Fig. 10) we display the reasoning of Qwen3-0.6B. While the content of segment ① at first sight looks relevant, closer inspection reveals that the final answer "80" is not at all deduced from the reasoning performed before. The answer seems to be a lucky guess, most likely informed by the fact that answers to such competition-level questions usually take the form of an integer value. This guessing is elegantly reflected in the potential curve; the reasoning in segment ① (which essentially encompasses the entire CoT) does not make any progress at all towards the final answer, precisely because the model is most likely making a guess in the end, which more often than not ends up being wrong.

6 Transferability of CoT Through the Lens of Potential

Motivated by the insights from Sec. 4 and 5, we now investigate if reasoning *insights* transfer between different families of models, which would further underscore that the mechanisms underlying CoT reasoning share parallels with human reasoning. We hypothesize that if the sub-steps present information gain through reasoning insights, (similar to e.g. Fig. 1), weaker models could be able to solve problems that were previously too difficult. We study this scenario for both reasoning and non-reasoning models. In the first setup we consider Qwen3-0.6B as the weak model, which is provided with partial CoT from its bigger version Qwen3-32B. We also explore traces from GPT-OSS-20B to

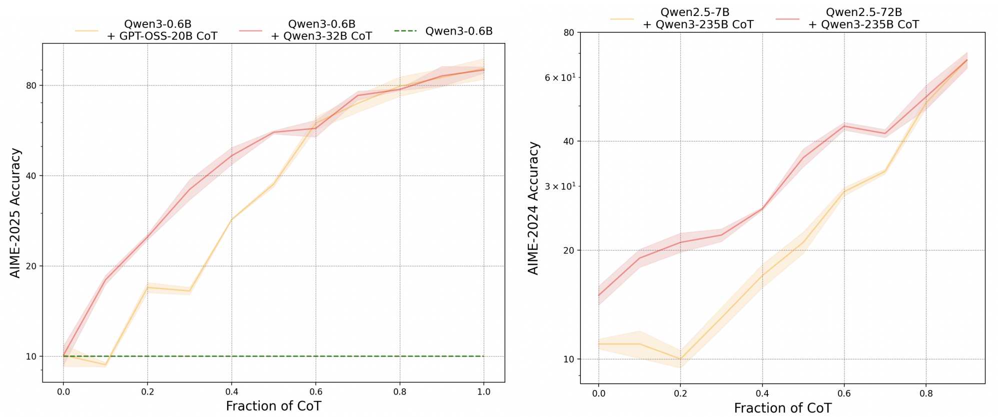

Figure 8: **Transferability of CoT. Left:** Accuracy on AIME-2025 of weaker reasoning model Qwen3-0.6B when provided with partial CoT from Qwen3-32B (red) and from GPT-OSS-20B (orange), leading to very quick improvements. **Right:** Accuracy of non-reasoning models Qwen2.5-7B and Qwen2.5-72B when provided with a partial CoT based on the final summary output of reasoning model Qwen3-235B.

further assess how robust transferability is with respect to out-of-distribution scenarios. For the non-reasoning models we instead create a dataset of *gold* CoT, using one of the strongest public models Qwen3-235B to produce answers on AIME-2024 in thinking mode. We then extract the CoT after thinking, which presents a clean summary of the long thinking traces and use these as partial traces. We then test the weaker Qwen2.5-7B and Qwen2.5-72B models, letting them complete the partial responses for various percentages. We display the resulting test accuracies as a function of the fraction of partial CoT in Fig. 8. We observe that surprisingly, in both reasoning and non-reasoning scenarios, the models manage to not only maintain their original accuracies but quickly improve (with as little as 20% CoT), answering previously unsolved questions. While the CoT does seem to transfer better within the same family, Qwen3-0.6B can still leverage the significantly different traces from GPT-OSS-20B, suggesting that the mechanisms driving the performance are universally shared between models to a strong degree.

7 Conclusion

In this work we have investigated chain-of-thought reasoning in large language models through the notion of the associated potential. We have performed an in-depth analysis of parts of CoT that strongly move the potential upwards (reasoning insights and jumps), as well as tokens that actively worsen the performance due to reasoning tangents. We further observed that especially for smaller LLMs, the potential can exhibit very late spikes only, suggesting that the final answer was reached without leveraging the reasoning. Upon qualitative examination we indeed found that many answers are guesses, leading to inflated $\text{pass@}k$ scores. We showed that more desirable potentials (free of tangents) can be obtained by an iterative procedure, resulting in more monotonic CoTs. Finally, we further investigate reasoning insights by introducing the notion of CoT transferability, which measures to what degree a weaker model can profit from the partial CoT of a stronger one. We show that the insights of the stronger model indeed help push the performance of the weaker one beyond what it can typically solve on its own, highlighting that CoT indeed relies on such interpretable mechanisms.

We believe that the potential can be used in future work to further understand how CoT contributes to successful completions of LLMs, helping to pin-point its important parts within the very long chains of today's reasoning models. The potential could also serve as a partial reward in RL training to perform more fine-grained credit assignment, indeed concurrent work has already started exploring this angle (Guo et al., 2025; Hou et al., 2025). Finally, we also hope that the transferability of partial CoTs can be leveraged in reinforcement learning to reduce sparsity of rewards with Amani et al. (2025) already obtaining positive initial results.

REFERENCES

Mohammad Hossein Amani, Aryo Lotfi, Nicolas Mario Baldwin, Samy Bengio, Mehrdad Farajtabar, Emmanuel Abbe, and Robert West. Rl for reasoning by adaptively revealing rationales, 2025. URL https://arxiv.org/abs/2506.18110.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Moshe Berchansky, Daniel Fleischer, Moshe Wasserblat, and Peter Izsak. CoTAR: Chain-of-thought attribution reasoning with multi-level granularity. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 236–246, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.13. URL https://aclanthology.org/2024.findings-emnlp.13/.

Siddhant Bhambri, Upasana Biswas, and Subbarao Kambhampati. Do cognitively interpretable reasoning traces improve llm performance?, 2025. URL https://arxiv.org/abs/2508.16695.

Eric J Bigelow, Ari Holtzman, Hidenori Tanaka, and Tomer Ullman. Forking paths in neural text generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8RCmNLeeXx.

Paul C. Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. Thought anchors: Which llm reasoning steps matter?, 2025. URL https://arxiv.org/abs/2506.19143.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

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 long reasoning models. In *Forty-second International Conference on Machine Learning*, 2025a. URL https://openreview.net/forum?id=MSbU3L7V00.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Samuel R. Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning models don't always say what they think, 2025b. URL https://arxiv.org/abs/2505.05410.

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

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models, 2023. URL https://arxiv.org/abs/2211.10435.

Olga Golovneva, Moya Peng Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. ROSCOE: A suite of metrics for scoring step-by-step reasoning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=xYlJRpzZtsY.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo

Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Yiran Guo, Lijie Xu, Jie Liu, Ye Dan, and Shuang Qiu. Segment policy optimization: Effective segment-level credit assignment in RL for large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=9osvTOYbT4.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL https://arxiv.org/abs/2103.03874.

Zhenyu Hou, Ziniu Hu, Yujiang Li, Rui Lu, Jie Tang, and Yuxiao Dong. TreeRL: LLM reinforcement learning with on-policy tree search. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12355–12369, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.604. URL https://aclanthology.org/2025.acl-long.604/.

Subbarao Kambhampati, Kaya Stechly, Karthik Valmeekam, Lucas Saldyt, Siddhant Bhambri, Vardhan Palod, Atharva Gundawar, Soumya Rani Samineni, Durgesh Kalwar, and Upasana Biswas. Stop anthropomorphizing intermediate tokens as reasoning/thinking traces!, 2025. URL https://arxiv.org/abs/2504.09762.

Vijay Konda and John Tsitsiklis. Actor-critic algorithms. In S. Solla, T. Leen, and K. Müller (eds.), *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999. URL https://proceedings.neurips.cc/paper_files/paper/1999/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning, 2023. URL https://arxiv.org/abs/2307.13702.

Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, Thomas Hubert, Peter Choy, Cyprien de Masson d'Autume, Igor Babuschkin, Xinyun Chen, Po-Sen Huang, Johannes Welbl, Sven Gowal, Alexey Cherepanov, James Molloy, Daniel J. Mankowitz, Esme Sutherland Robson, Pushmeet Kohli, Nando de Freitas, Koray Kavukcuoglu, and Oriol Vinyals. Competition-level code generation with alphacode. *Science*, 378(6624):1092–1097, 2022. doi: 10.1126/science.abq1158. URL https://www.science.org/doi/abs/10.1126/science.abq1158.

Anton Lozhkov, Raymond Li, Loubna Ben Allal, Federico Cassano, Joel Lamy-Poirier, Nouamane Tazi, Ao Tang, Dmytro Pykhtar, Jiawei Liu, Yuxiang Wei, Tianyang Liu, Max Tian, Denis Kocetkov, Arthur Zucker, Younes Belkada, Zijian Wang, Qian Liu, Dmitry Abulkhanov, Indraneil Paul, Zhuang Li, Wen-Ding Li, Megan Risdal, Jia Li, Jian Zhu, Terry Yue Zhuo, Evgenii Zheltonozhskii, Nii Osae Osae Dade, Wenhao Yu, Lucas Krauß, Naman Jain, Yixuan Su, Xuanli He, Manan Dey, Edoardo Abati, Yekun Chai, Niklas Muennighoff, Xiangru Tang, Muhtasham Oblokulov, Christopher Akiki, Marc Marone, Chenghao Mou, Mayank Mishra, Alex Gu, Binyuan Hui, Tri Dao, Armel Zebaze, Olivier Dehaene, Nicolas Patry, Canwen Xu, Julian McAuley, Han Hu, Torsten Scholak, Sebastien Paquet, Jennifer Robinson, Carolyn Jane Anderson, Nicolas Chapados, Mostofa Patwary, Nima Tajbakhsh, Yacine Jernite, Carlos Muñoz Ferrandis, Lingming Zhang, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. Starcoder 2 and the stack v2: The next generation, 2024. URL https://arxiv.org/abs/2402.19173.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, Yansong Tang, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct, 2025. URL https://arxiv.org/abs/2308.09583.

Thang Luong, Dawsen Hwang, Hoang H. Nguyen, Golnaz Ghiasi, Yuri Chervonyi, Insuk Seo, Junsu Kim, Garrett Bingham, Jonathan Lee, Swaroop Mishra, Alex Zhai, Clara Huiyi Hu, Henryk Michalewski, Jimin Kim, Jeonghyun Ahn, Junhwi Bae, Xingyou Song, Trieu H. Trinh, Quoc V. Le, and Junehyuk Jung. Towards robust mathematical reasoning, 2025. URL https://arxiv.org/abs/2511.01846.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. Faithful chain-of-thought reasoning. In Jong C. Park, Yuki Arase, Baotian Hu, Wei Lu, Derry Wijaya, Ayu Purwarianti, and Adila Alfa Krisnadhi (eds.), *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 305–329, Nusa Dua, Bali, November 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.ijcnlp-main.20. URL https://aclanthology.org/2023.ijcnlp-main.20/.

MAA. American invitational mathematics examination, 2025. URL https://maa.org/maa-invitational-competitions/. Accessed: 2025-08-08.

Aman Madaan and Amir Yazdanbakhsh. Text and patterns: For effective chain of thought, it takes two to tango, 2022. URL https://arxiv.org/abs/2209.07686.

Aman Madaan, Katherine Hermann, and Amir Yazdanbakhsh. What makes chain-of-thought prompting effective? a counterfactual study. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023a. URL https://openreview.net/forum?id=va7nzRsbA4.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback, 2023b. URL https://arxiv.org/abs/2303.17651.

Andreas Madsen, Sarath Chandar, and Siva Reddy. Are self-explanations from large language models faithful? In *Annual Meeting of the Association for Computational Linguistics*, 2024. URL https://api.semanticscholar.org/CorpusID:266999774.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024. URL https://arxiv.org/abs/2412.16720.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models, 2023. URL https://arxiv.org/abs/2210.03350.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

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 qa benchmark, 2023. URL https://arxiv.org/abs/2311.12022.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code, 2024. URL https://arxiv.org/abs/2308.12950.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023. URL https://arxiv.org/abs/2303.11366.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity, 2025. URL https://arxiv.org/abs/2506.06941.

Kaya Stechly, Karthik Valmeekam, Atharva Gundawar, Vardhan Palod, and Subbarao Kambhampati. Beyond semantics: The unreasonable effectiveness of reasonless intermediate tokens, 2025. URL https://arxiv.org/abs/2505.13775.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018. ISBN 0262039249.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2717–2739, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.153. URL https://aclanthology.org/2023.acl-long.153/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Skyler Wu, Eric Meng Shen, Charumathi Badrinath, Jiaqi Ma, and Himabindu Lakkaraju. Analyzing chain-of-thought prompting in large language models via gradient-based feature attributions, 2023. URL https://arxiv.org/abs/2307.13339.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WZH7099tgfM.

A Appendix

A.1 Experimental Details

We use vllm (Kwon et al., 2023) for all of our experiments. For potential calculation we set $N = 128$ and use a temperature of $T = 0.6$ and $p = 0.95$ as sampling parameters. For all models and datasets we generate $T = 32k$ tokens excluding the prompt. To ensure that the potential does not increase due to higher generation length, we always subtract the length of the partial CoT from $32k$ and use this number as T.

A.2 Proof of Proposition 1

Here we present the previously omitted proof of Proposition

By Bayes' rule, for any token c_{t+1} we have

$$f_{t+1} = \mathbb{P}(y = 1 \mid x, c_{1:t}, c_{t+1}) = \frac{f_t\, p_1(c_{t+1})}{f_t\, p_1(c_{t+1}) + (1 - f_t)\, p_0(c_{t+1})}.$$

Taking expectation with respect to c_{t+1} drawn from p_1, i.e. conditioned on the event that the rest of the run is correct, gives

$$\mathbb{E}[f_{t+1}] = f_t \sum_{c_{t+1}} p_1(c_{t+1}) \frac{p_1(c_{t+1})}{f_t p_1(c_{t+1}) + (1 - f_t) p_0(c_{t+1})}.$$

Equivalently,

$$\mathbb{E}[f_{t+1}] = f_t \sum_{c_{t+1}} \frac{p_1(c_{t+1})^2}{f_t p_1(c_{t+1}) + (1 - f_t) p_0(c_{t+1})}.$$

Now apply the Cauchy–Schwarz inequality with weights $q(c_{t+1}) = f_t p_1(c_{t+1}) + (1 - f_t) p_0(c_{t+1})$:

$$\left(\sum_{c_{t+1}} \frac{p_1(c_{t+1})^2}{q(c_{t+1})} \right) \left(\sum_{c_{t+1}} q(c_{t+1}) \right) \geq \left(\sum_{c_{t+1}} p_1(c_{t+1}) \right)^2.$$

Since $\sum_{c_{t+1}} q(c_{t+1}) = 1$ and $\sum_{c_{t+1}} p_1(c_{t+1}) = 1$, it follows that

$$\sum_{c_{t+1}} \frac{p_1(c_{t+1})^2}{q(c_{t+1})} \geq 1.$$

Therefore,

$$\mathbb{E}[f_{t+1}] \geq f_t.$$

Finally, taking expectation over prefixes $c_{1:t}$ distributed as on correct runs yields

$$\mathbb{E}[f_{t+1}] \geq \mathbb{E}[f_t],$$

which is the desired result.

A.3 More CoT Examples

We display more quantitative results in Table 2 for the MATH-500 dataset, in Table 3 we show results for the coding benchmark HumanEval (Chen et al., 2021) and in Table 4 the ones for GPQA-Diamond (Rein et al., 2023). We find that the same trends observed for AIME-2024 hold also in this case. Interestingly, models seem more stable on coding benchmarks, with their CoT displaying more monotonic behaviour and in general less tangents. For GPQA we find trends more consistent with AIME, most likely because more questions are very difficult compared to HumanEval. As expected due to GPQA being a multiple choice benchmark with only four choices, we observe a significantly higher guessing rate. In this case we adjusted the threshold for guessing to 25% (as opposed to the previously used 5%) as a random guessing baseline would of course always achieve 25%.

We also show more annotated CoT in Fig. 7, Fig. 9 and for coding in Fig. 14 and Fig. 15. In Fig. 9 we have again have the model performing a reasoning insight, correctly realizing that the exponents can be deduced from the binary representation of the number. We then finally have a reasoning jump, where the model experiences a strong boost in potential from the word "correspond". While at first sight not clearly interpretable, we hypothesize that this word forces the model to output concrete values for a_i's, otherwise a common failure model as the model tries to further refine their computation.

Model	Reasoning	Insights ↑	Tangents ↓	Late spike	Monotonicity
Qwen2.5-1.5B	✗	23%	11%	1%	40%
Qwen2.5-7B	✗	24%	12%	1%	41%
Llama3.1-8B	✗	19%	21%	1.8%	31%
Llama3.1-70B	✗	22%	23%	1.5%	30%
Qwen3-0.6B	✓	30%	40%	1.4%	30%
Qwen3-32B	✓	28%	35%	0.9%	45%

Table 2: Behaviours of potential for several reasoning and non-reasoning models on MATH-500.

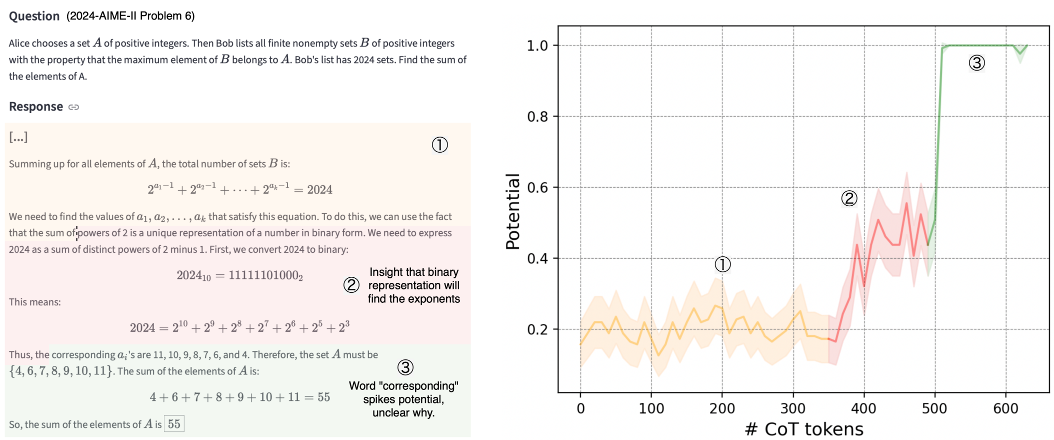

Figure 9: **Unintuitive reasoning jumps.** Qwen2.5-7B's potential $\mathrm{pot}_{256}(\bullet; \boldsymbol{x})$ remains flat in ① although crucial insights are obtained. The potential then increases due to a reasoning *insight* in ② (realizing that the binary representation determines the exponents). In ③ we obtain the final spike at the word "corresponding", a reasoning *jump*, which seems strange from a human perspective. We hypothesize that it might force the model to output values for a_i's, which indeed is the next logical step. We indeed observe that without this word, the model continues to perform unnecessary calculations, subsequently leading to wrong values for a_i.

Model	Reasoning	Insights ↑	Tangents ↓	Late spike	Monotonicity
Qwen2.5-1.5B	✗	39%	7%	0%	55%
Qwen2.5-7B	✗	30%	4.8%	0%	73%
Llama-3.1-8B	✗	41%	16.5%	0.4%	42.1%

Table 3: Behaviours of potential for several reasoning and non-reasoning models on the coding benchmark HumanEval.

Model	Reasoning	Insights ↑	Tangents ↓	Late spike	Monotonicity
Qwen2.5-1.5B	✗	17%	6.8%	19%	20.9%
Llama-3.1-8B	✗	29.1%	20.0%	17.6%	25.8%

Table 4: Behaviours of potential for several reasoning and non-reasoning models on the coding benchmark GPQA-Diamond.

In Fig. 10 we again observe a reasoning guess from Qwen2.5-1.5B, where the CoT in segment ①, while seemingly making sense at first sight, actually does not contribute to the final answer at all. In fact the number 80 does not relate at all to the computations made before. This is reflected in the potential, that shows a spike only towards the very end, highlighting that the CoT indeed did not contribute.

Finally, we show an instance of optimized CoT introduced in Sec.4. We observe that the potential is now strongly monotonic, with almost every partial CoT leading to some improvement in the potential. This is also reflected qualitatively, we can see that the CoT is more concise in language, in fact we can display all of it here. In segment ① the model makes slower progress as those are steps it can reliably do. Finally, the model undergoes a reasoning insight ② with the model discovering that d needs to divide 56.

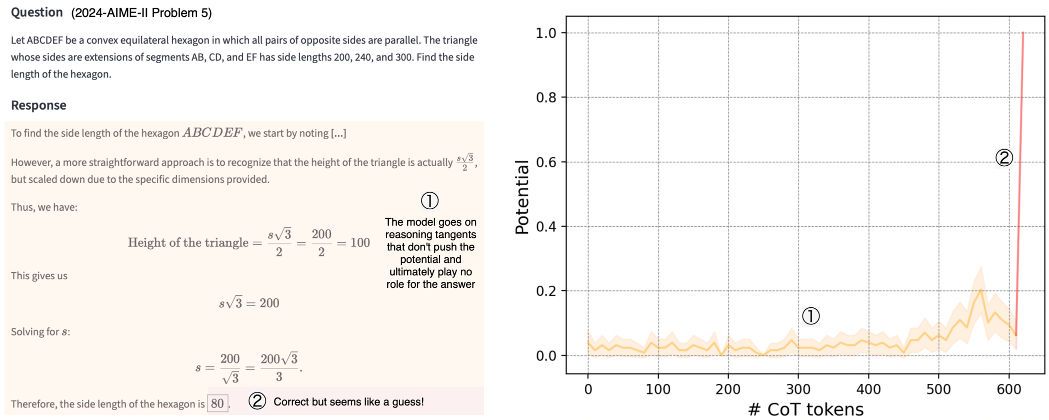

Figure 10: **Reasoning tangents and guessing.** Qwen2.5-1.5B goes on a long reasoning tangent in ① that does not increase the potential over a long token horizon. Finally it outputs a final answer in ② unrelated to the previous reasoning that happens to be correct.

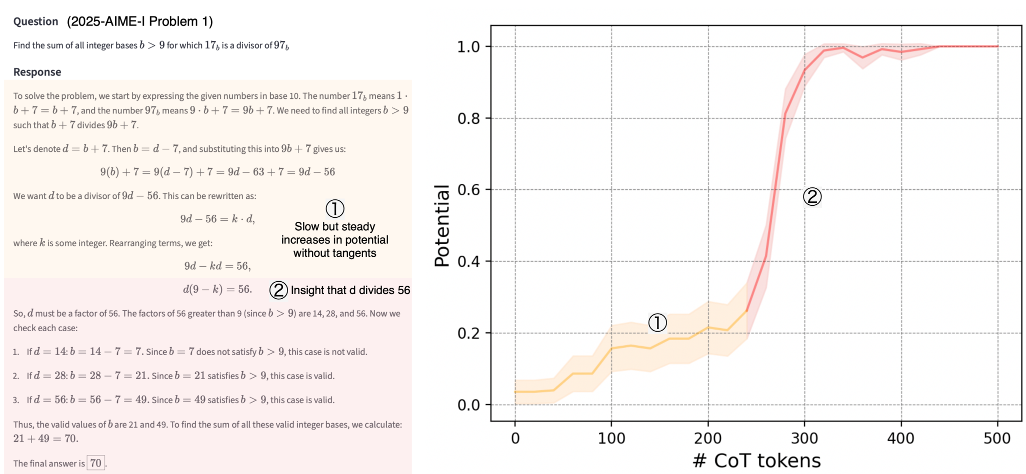

Figure 11: **Optimized CoT.** We show a trajectory based on the optimized CoT from Qwen2.5-1.5B. The CoT is more concise, actually allowing us to show it here in full length. The potential is monotonic as anticipated and all tokens contribute to it. In segment ① the model makes slower progress as those are steps it can reliably do. Finally, the model undergoes a reasoning insight ② with the model discovering that d needs to divide 56.

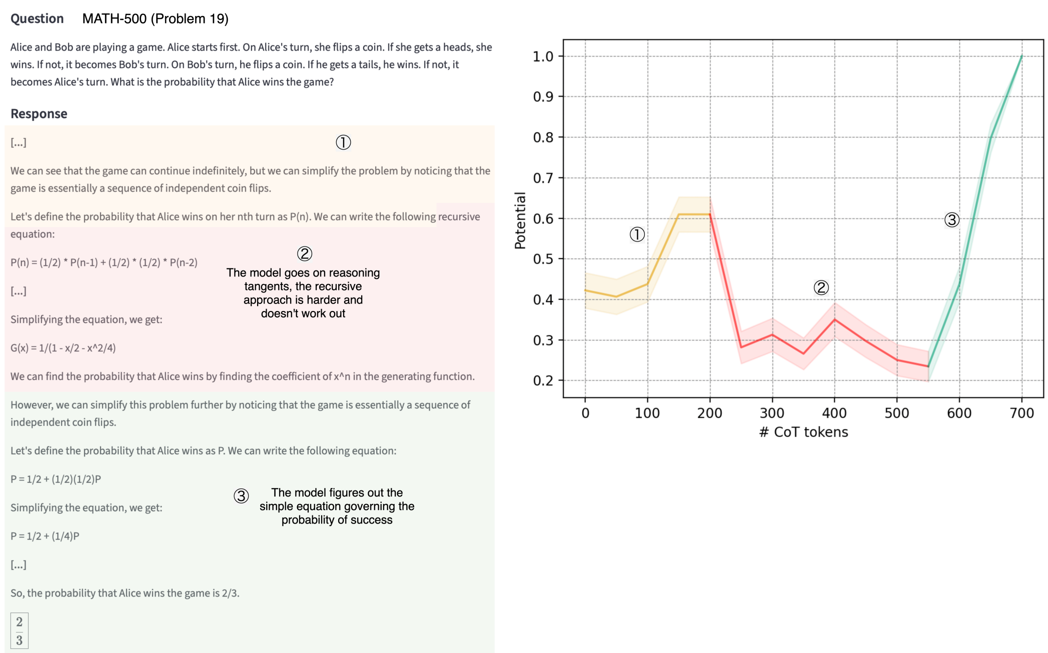

Figure 12: **Reasoning tangents and insights for Llama3.1-8B on Math-500.** We show an example for another dataset using Llama3.1-8B. Similar observations hold; the models tend to go on reasoning tangents as shown here with the model exploring a more difficult recursive approach, which it ultimately abandons by deriving the key insight, expressing the probability as a simple linear equation.

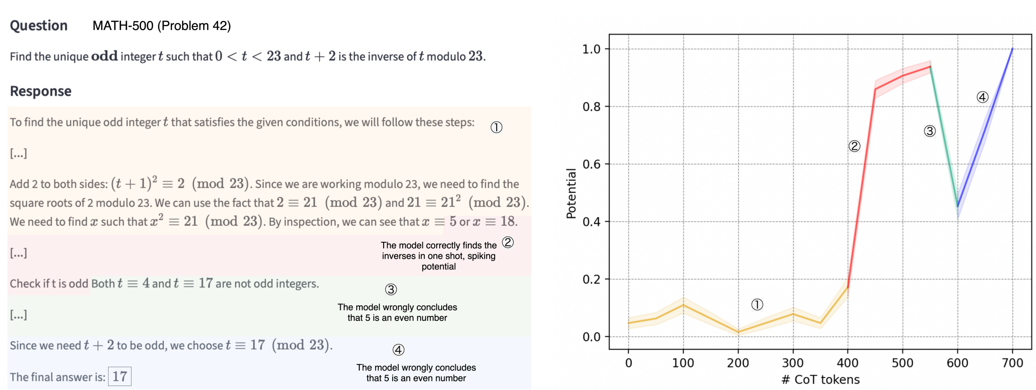

Figure 13: **Reasoning tangents and insights for Llama3.1-8B on Math-500.** We show an example for another dataset using Llama3.1-8B. Again the model displays a key insight, realizing directly that 5 and 18 are the square roots. A small tangent is also encountered, with the model missing that 17 is an odd number, but subsequently corrected later in the reasoning.

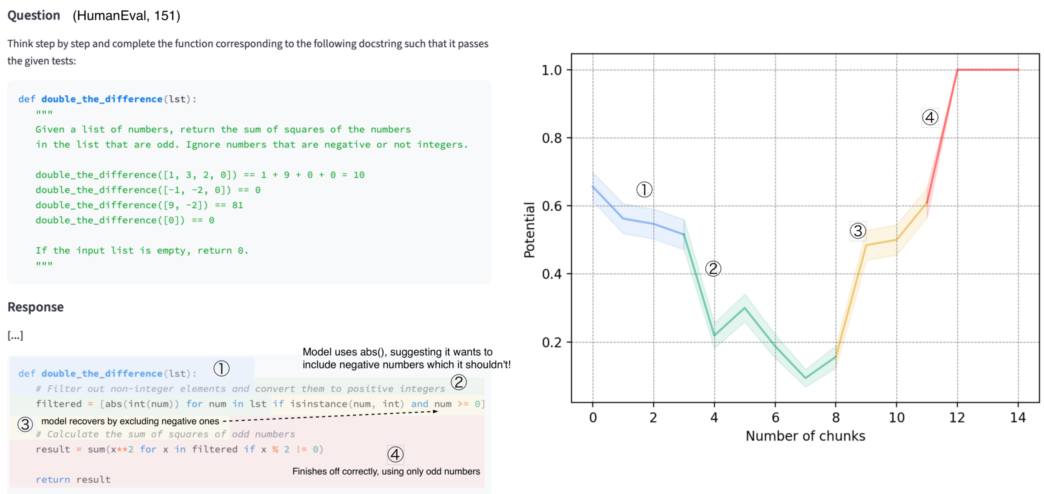

Figure 14: **Potential for coding tasks.** We analyze a trace of Qwen2.5-1.5-Instruct on HumanEval and identify the occurence of tangents and insights. Here, the model gets distracted initially by outlining that it will convert all numbers to positive integers, and indeed using commands abs() and int(), dropping the potential significantly as a consequence. The model then recovers by fixing the for-loop using if statements, which actually render abs() and int() effectless as the loop anyways only goes over positive integers.

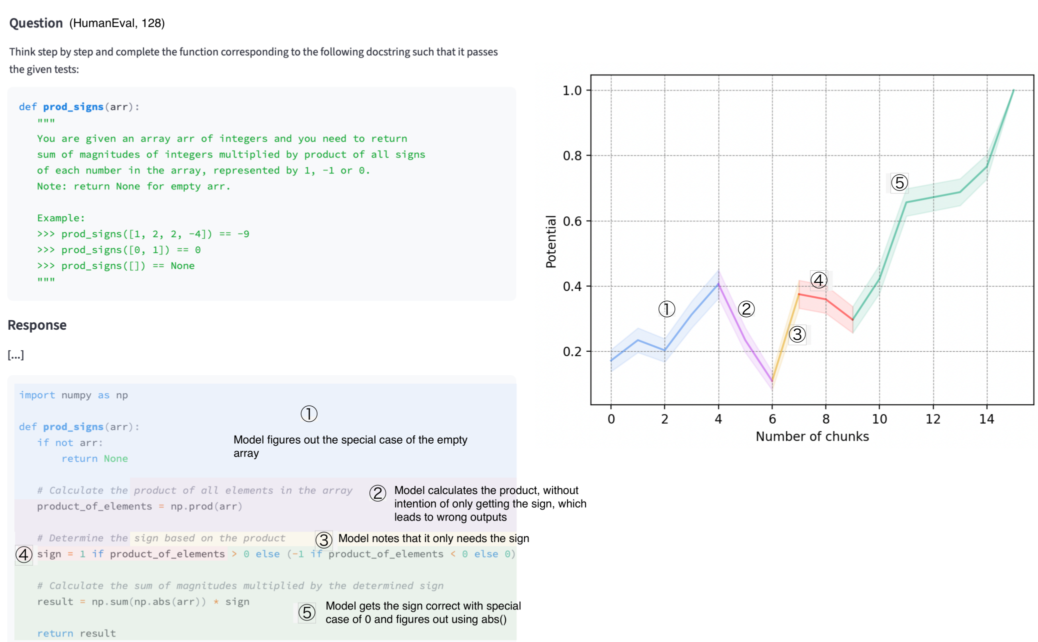

Figure 15: **Potential for coding tasks.** We analyze a trace of Qwen2.5-1.5-Instruct on HumanEval and identify the occurence of tangents and insights.

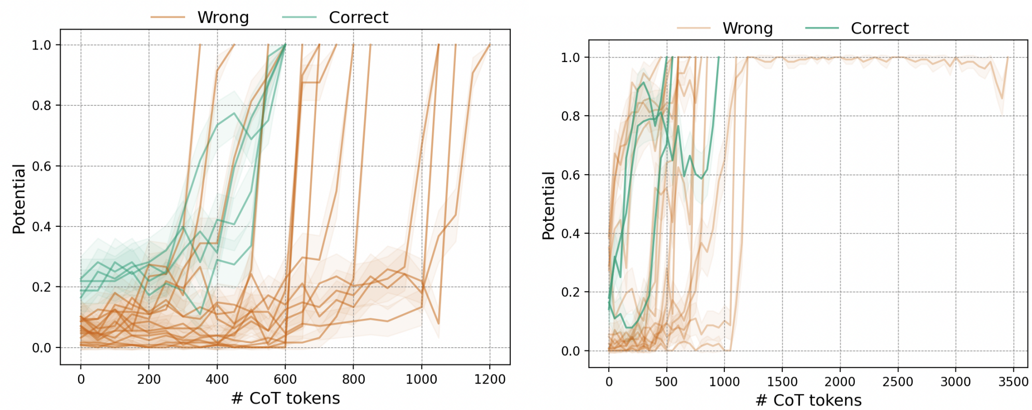

Figure 16: **Stability profiles.** Stability profiles for Qwen2.5-1.5B and Qwen2.5-7B on AIME 7 and 26 respectively. Correct and wrong answers exhibit similar profiles across models and questions.

A.4 Stability of CoT

We can also consider a slight variation of the potential, called the stability of a CoT. Given a prompt $\boldsymbol{x}$, CoT reasoning and answer pair $(\boldsymbol{c}, y)$ we define the stability of a sub-chain $\boldsymbol{c}_{<t}$ as

$$\text{stable}_N(\boldsymbol{c}_{<t}; \boldsymbol{x}, y) := \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}_{\{y^{(n)}=y\}} \quad \text{where } \left(y^{(n)}, \boldsymbol{c}_{\geq t}^{(n)}\right) \sim \texttt{LM}_\theta(\bullet|\boldsymbol{c}_{<t}, \boldsymbol{x})$$

with the slight variation that instead of considering the ground truth y^*, we now consider the reached final answer of the chain $\boldsymbol{c}$ as the target. I.e. the potential is a special of stability, when $y = y^*$. Stability measures how *determined* the final answer is throughout the reasoning process of the model. Somewhat surprisingly, we observe that correct answers do not necessarily always display higher stability, indicating that models can become convinced very early on in their reasoning about wrong answers. We display various stability curves in Fig. 16.

A.5 Computational complexity of potential calculation

The computational load to obtain the potential curve of a given prompt scales with the number of evaluations N produced to estimate the potential and the number of points N_{chunks} we calculate the potential for. More precisely, say for a generation from scratch we produce T tokens, the total amount of tokens produced to estimate the potential curve will be given by

$$T_{\text{tot}} = N \sum_{i=1}^{N_{\text{chunks}}} \frac{i}{N_{\text{chunks}}} T \approx \frac{N N_{\text{chunks}} T}{2}$$

N_{chunks} hence also gives us a way to control the amount of compute and we set it to moderate values of $N_{\text{chunks}} = 15$ for the quantitative evaluations, while we use higher values of around $N_{\text{chunks}} = 50$ for most figures, to obtain a more fine-grained understanding. Advanced inference techniques such as prefix-caching and continuous batching in vLLM further help to speed up the potential calculation, as all of the prefill is shared among the prompts, and every prompt can in theory be generated in parallel, thus allowing for continuous batching. This is different for the optimal CoT, where inference has to be done sequentially as things depend on eachother, making it significantly slower.

A.6 Calculation of Optimal CoT

We detail the recipe to calculate the optimal CoT in Algorithm 1 below:

Algorithm 1 Generating potential-optimized CoTs

1: Initialize the CoT $\boldsymbol{c}_{<t} \leftarrow \emptyset$
2: **while** the chosen candidate does not contain the answer **do**
3: Sample M candidate CoT chunks $\boldsymbol{c}_{t:(t+T)}^{(m)} \overset{i.i.d.}{\sim} \mathtt{LM}_\theta(\cdot \mid \boldsymbol{c}_{<t}, \boldsymbol{x})$ of length C, for $m = 1, \dots, M$
4: Compute potentials $p_m \leftarrow \mathrm{pot}_N(\boldsymbol{c}_{<t+T}^{(m)}; \boldsymbol{x})$
5: Select $\tilde{m} \leftarrow \arg\max_m p_m$
6: Update $\boldsymbol{c}_{<t+T} \leftarrow [\boldsymbol{c}_{<t}, \boldsymbol{c}_{t<(t+T)}^{(\tilde{m})}]$
7: **end while**

The computational complexity of the optimal CoT calculation is quite high with $\approx \frac{MNN_{\text{chunks}}T}{2}$ tokens needed, while also not enjoying the same benefits from prefix caching and continuous batching, as the potential curve calculation did. We thus do not recommend its usage in practical setting but view it more as a proof of concept.

A consequence of our greedy strategy is the following: The optimal CoT could degenerate to a ”lucky guessing” answer when M goes to infinity, given that the model has non-zero support across all tokens, as this would effectively minimize the objective. Since we are restricted to using a finite M however in practice, we are implicitly regularizing the objective to sequences of sufficient probability mass, naturally circumventing this degeneracy. To make this regularization more explicit, one could potentially enhance the objective with a regularization term to require a given chain to meet a certain probability budget, eliminating this possibility. We observe however that M being finite is enough to avoid such degenerate cases.
We do however want to stress that the optimal CoT does depend on M.

A.7 More details on summary statistics

Here we provide the definitions for the statistics we used in Sec. 4. In all experiments we divide the CoT into 20 chunks, getting thus potential curves consisting of 20 points.

- **Insight:** We say that a given potential contains an insight if the difference between two consecutive chunks of CoT exceeds 40%, i.e. if one step of CoT raised the potential by at least 40%. We exclude the last two steps to make sure we don't count the late reasoning spikes as insights.
- **Tangent:** We define a potential to exhibit a tangent if the potential drops by at least 30%, not necessarily consecutively.
- **Guess:** We define late reasoning spikes or guesses as the case when the potential at the second to last step is smaller than 5%.
- **Monotonicity:** We call a potential monotone if its consecutive steps do not decrease by more than 10%.