Best-of-∞ — Asymptotic Performance of Test-Time Compute

RIKEN AIP, LLM×ML Workshop, Oct 3 2025

Junpei Komiyama (MBZUAI / NYU / RIKEN AIP),
joint work with Daisuke Oba (Sci Tokyo), Masafumi Oyamada (NEC)

(This slide was heavily assisted by LLMs, though the best-of-N strategy was not used.)

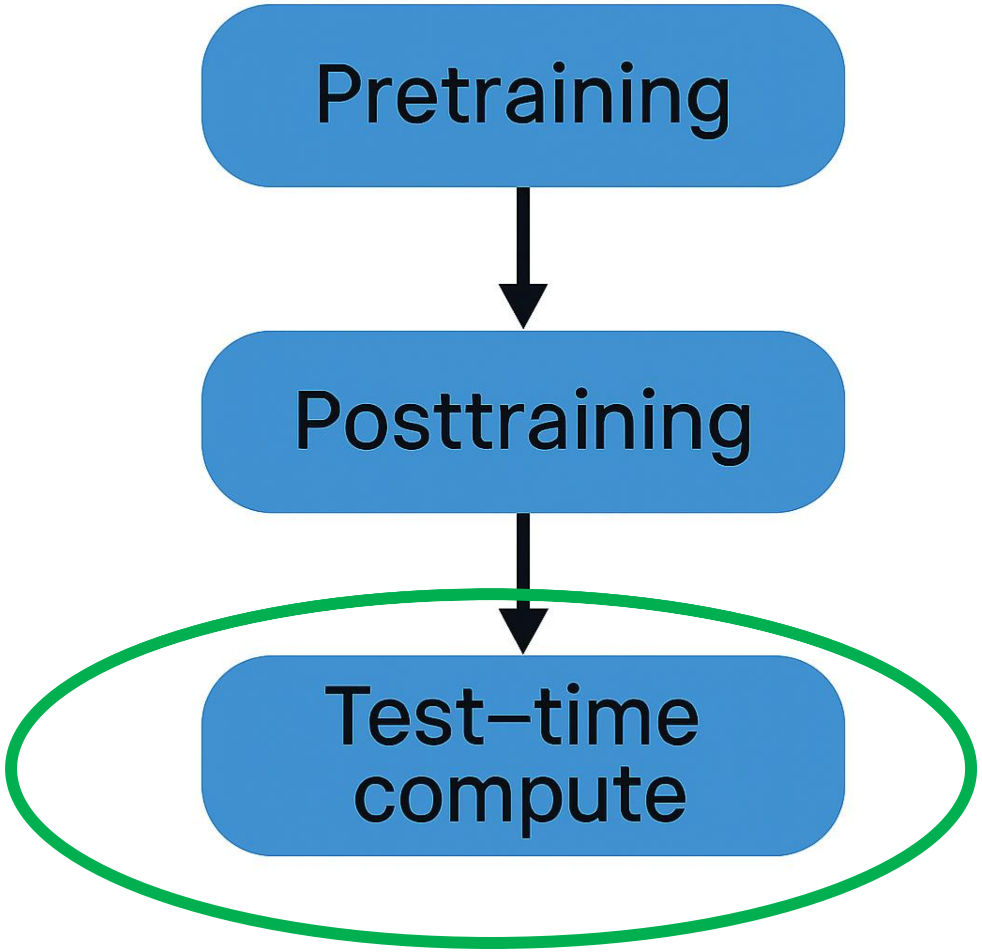
Agenda

- Introduction
  - Best-of-∞, the limit of Best-of-N with large N
- Adaptive Sampling:
  - Efficient test-time compute, approaching Best-of-∞ with limited computational budget
- LLM Ensemble
  - Complementary strength of multiple LLMs
- Summary

Test-time compute

- This talk focuses on test-time compute, where we use LLMs are used to answer questions.

THIS TALK IS ABOUT →



LLM on Math Reasoning

- AIME2024 (high-school math competition, pre-Olympiad level):

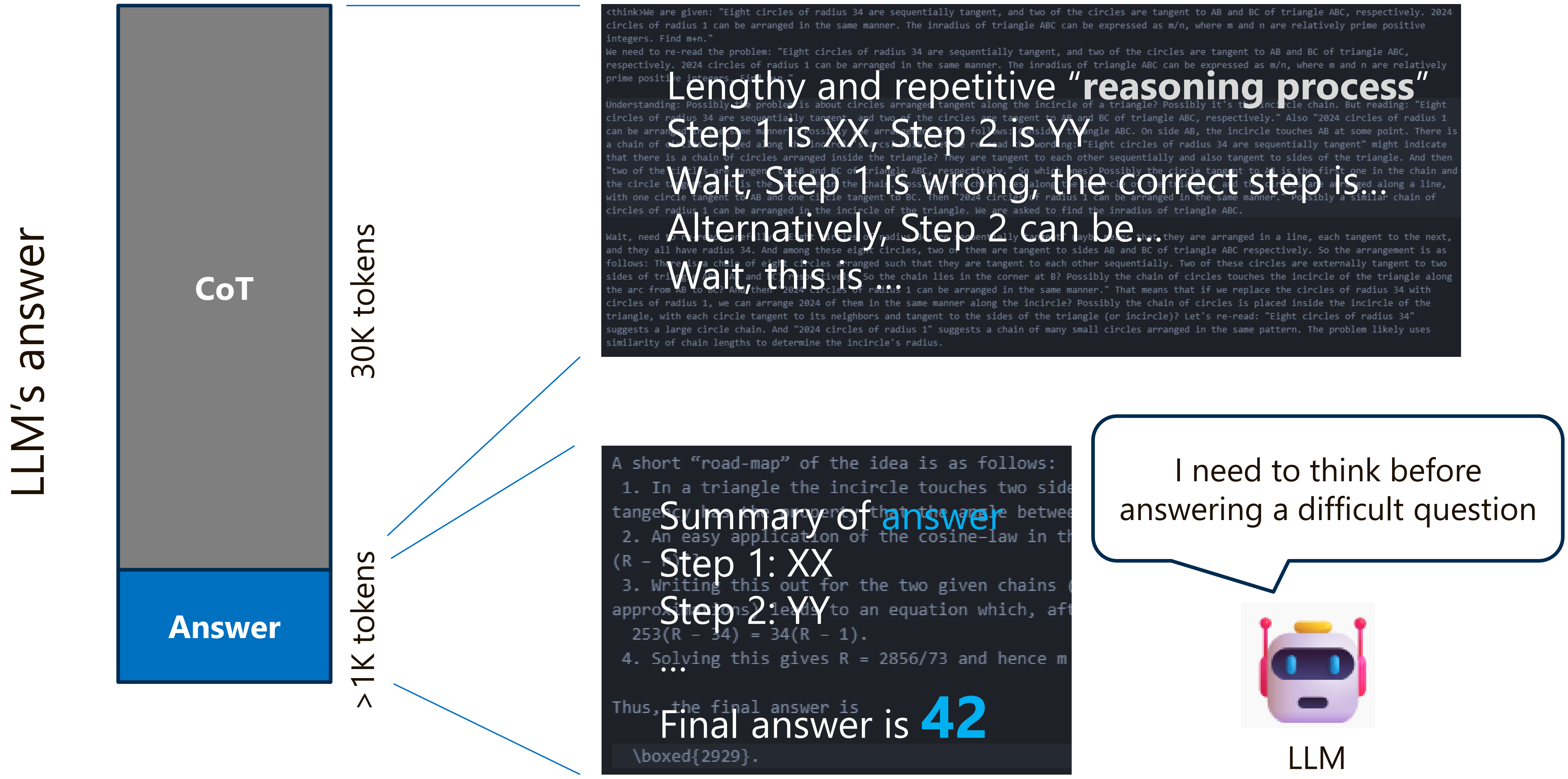
Q. Eight circles of radius 34 are sequentially tangent, and two of the circles are tangent to sides AB and BC of triangle ABC, respectively.
2024 circles of radius 1 can be arranged in the same manner.
The inradius of triangle ABC can be expressed as $\frac{m}{n}$,
where m and n are relatively prime positive integers.
Find $m + n$.

Challenging math question:
And I ask a local LLM
(say, gpt-oss-20b, phi-4-reasoning)
for answering this question.

Goal

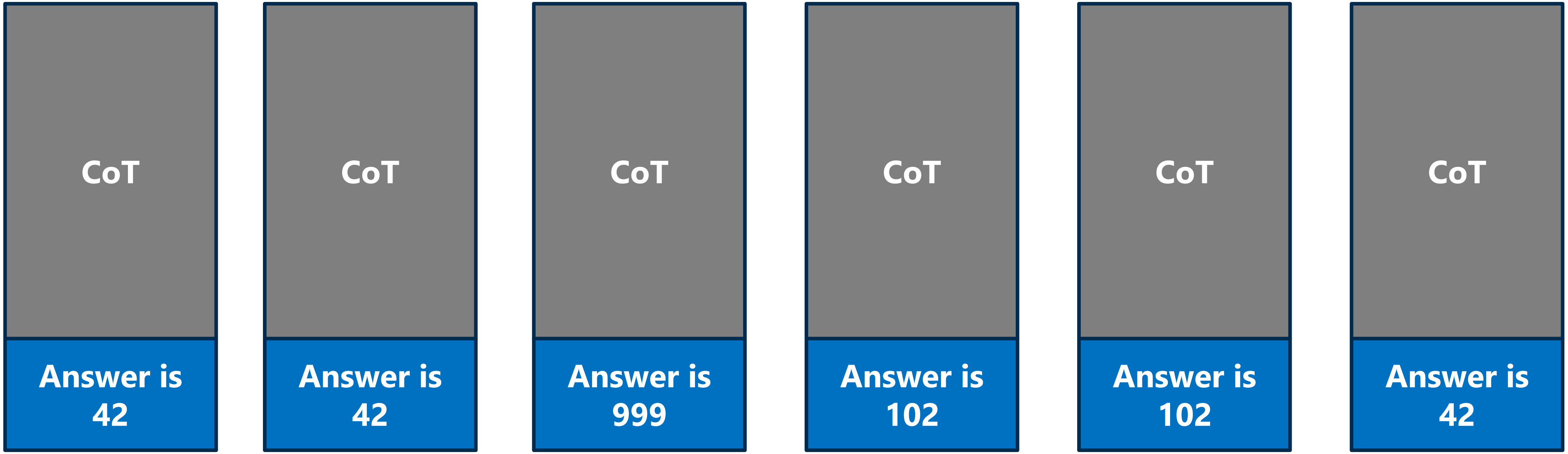
- Math/sci reasoning tasks (e.g., AIME2024/2025, MATH500, GPQA-DIAMOND)
- If we have access to infinite test-time compute resource (= local LLMs), how accurate we can be, in answering such questions?
  - We define Best-of-∞ performance.
- How close we can be, given a finite test-time compute budget?
  - Use computational resources efficiently.

LLM uses Chain-of-thoughts (CoT) before answering



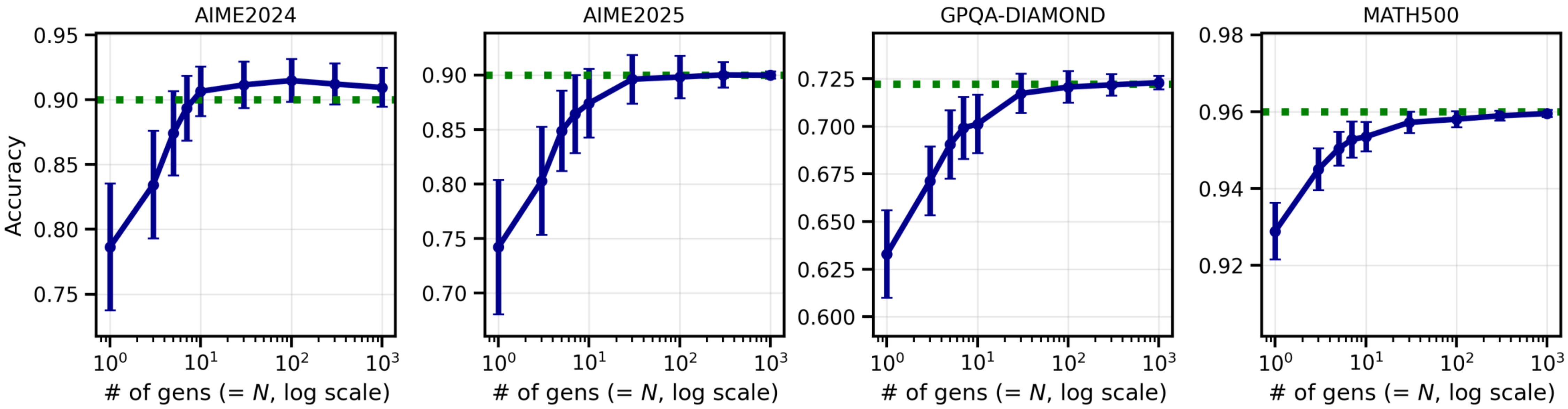
LLM reasoning: Best-of-N

- LLM's decoding process is random. Sometimes it answers correctly/incorrectly.
- Use best-of-N: Generate N answers.
- Selection rule: Majority voting.



Best-of-N improves accuracy

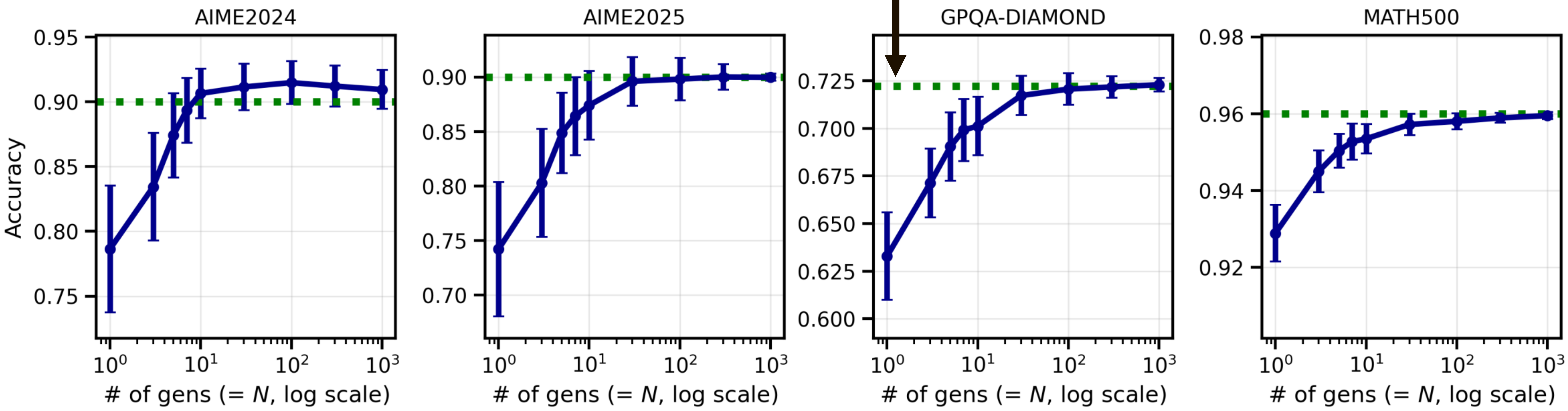
- Larger N, better accuracy.
  - Effective till N=100~1000.



Best-of-N improves accuracy

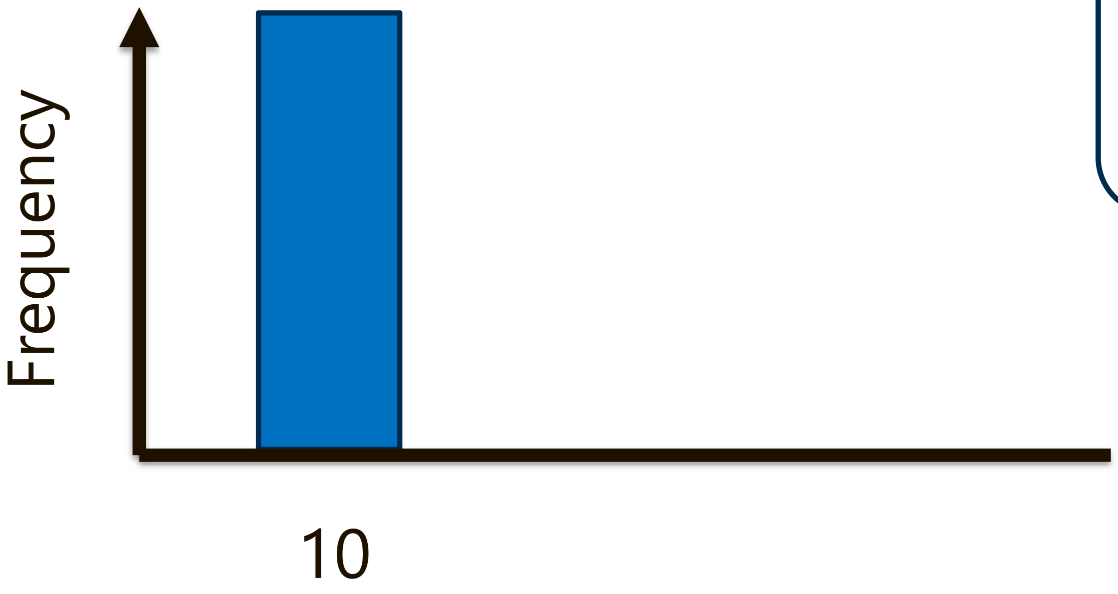
- Larger N, better accuracy.
  - Effective till N=100~1000.

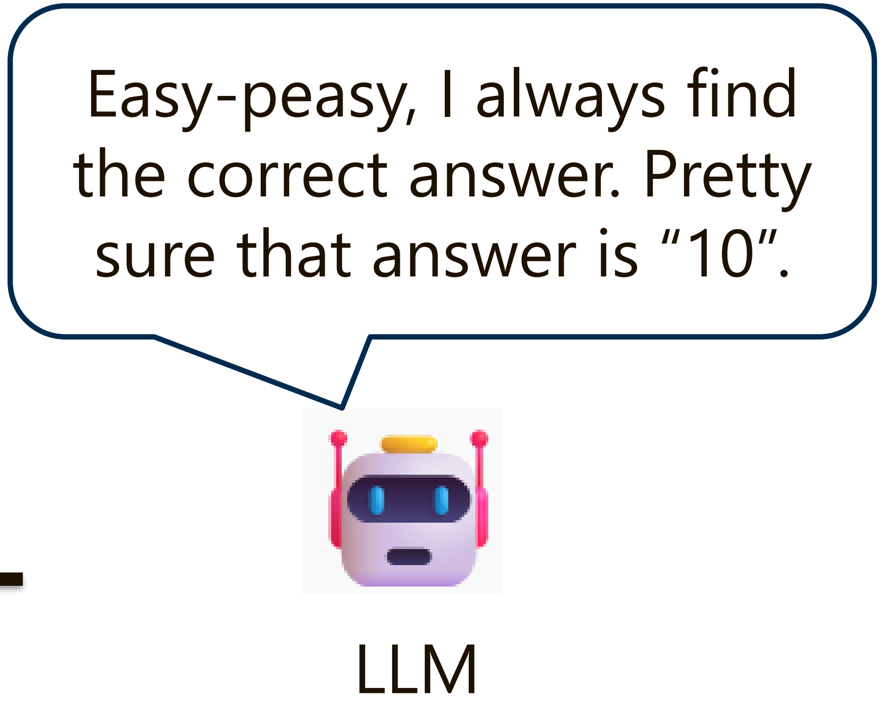
Green dotted line
= Best-of-∞ performance



Best-of-∞ Performance

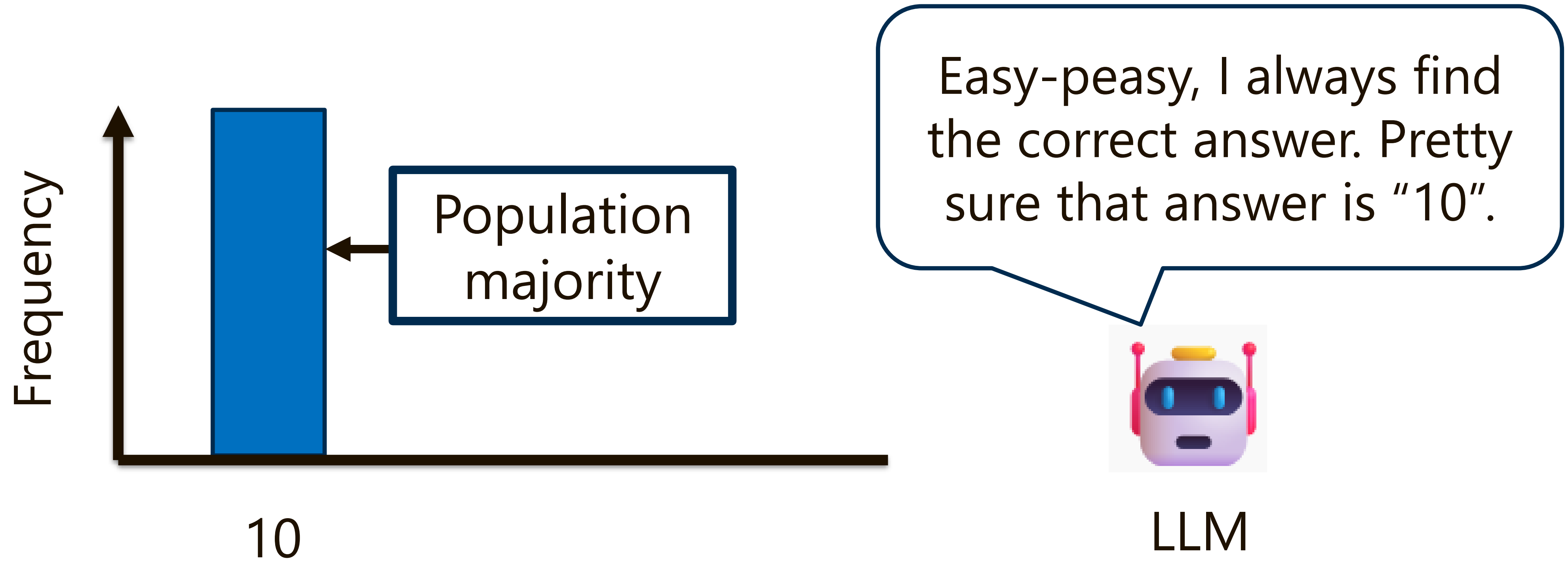
- At the limit of $N \to \infty$, there is a distribution of answers.
- For an **easy** question, LLM's answer is uniform (and correct)





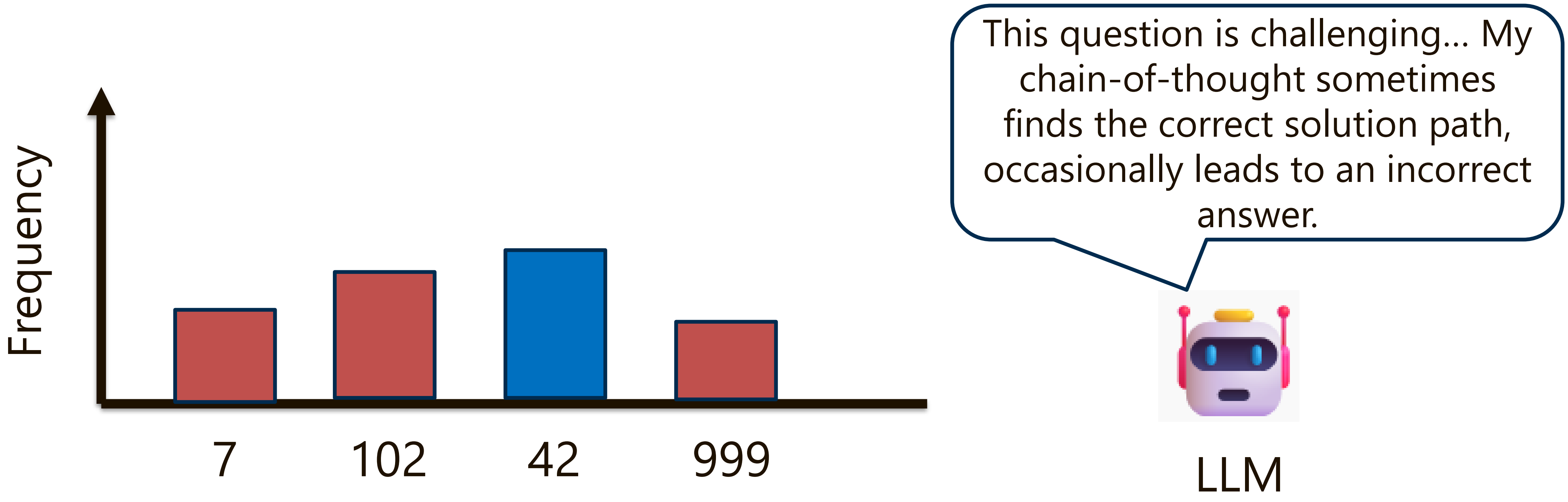
Best-of-∞ Performance

- At the limit of $N \to \infty$, there is a distribution of answers.
- For an **easy** question, LLM's answer is uniform (and correct)



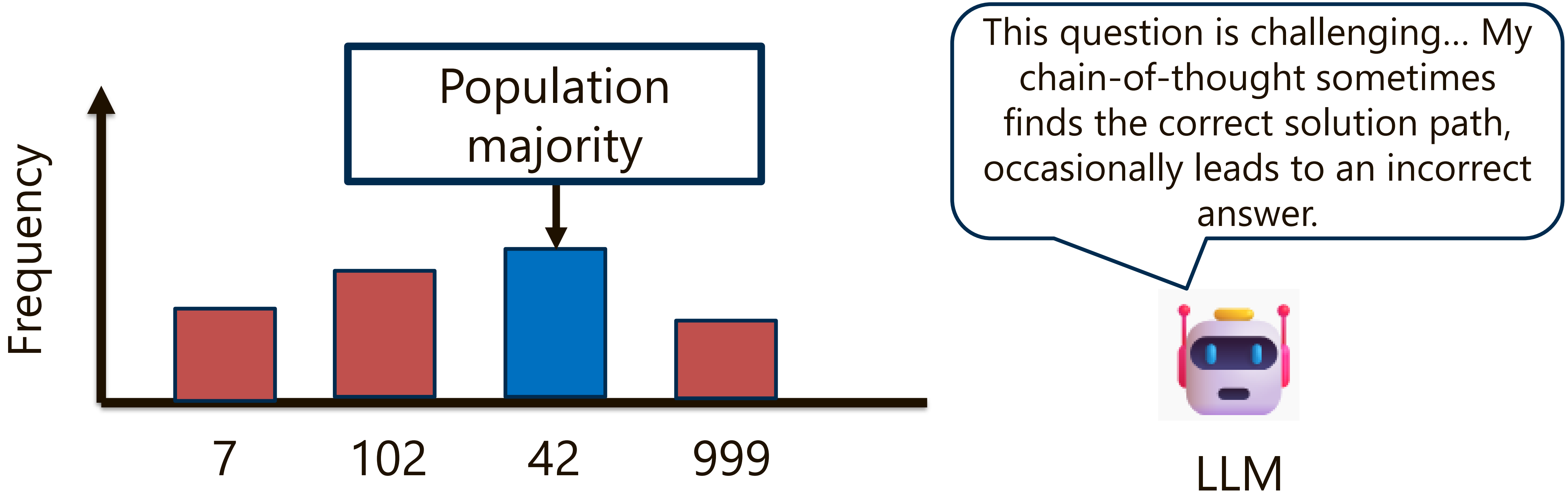
Best-of-∞ Performance

- At the limit of $N \to \infty$, there is a distribution of answers.
- For a **hard** question, LLM's answer varies



Best-of-∞ Performance

- At the limit of $N \to \infty$, there is a distribution of answers.
- For a **hard** question, LLM's answer varies



Best-of-∞ Performance

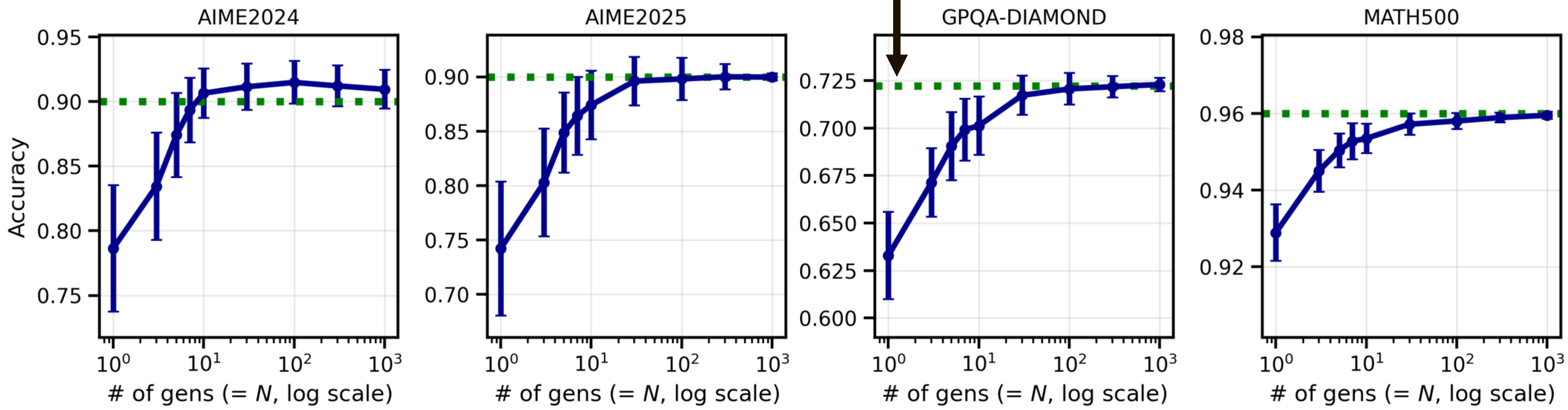
- Best-of-∞ performance is defined as

$$\frac{1}{|\mathrm{Questions}|} \sum_{q \in \mathrm{Questions}} 1[\mathrm{Gold\ answer} = \mathrm{population\ majority\ answer}]$$

- For example, AIME2025 consists of 30 questions. Phi-4-reasoning (Microsoft's open weight LLM)'s population majority matches the gold answer in 25 out of 30 questions.
  - The Best-of-∞ performance is 0.833 (= 25/30)

Best-of-∞ accuracy

To achieve Best-of-∞ accuracy, we need infinite, or sufficiently large N, samples



Adaptive Sampling

- Adaptive sampling: Ask LLM for the next generation or terminate.
- Better use of test-time compute budget:
  - For easy problems -> consistent answers -> early termination
  - For hard problems -> answer varies -> ask LLM for more answers
- Demo: https://jkomiyama.github.io/bestofinfty/llm-consensus-demo.html

Adaptive Sampling: When to terminate?

- Consider it as a hypothesis testing

$$H_0 : \text{The most frequent answer } A_1 \text{ is not the true majority.}$$
$$H_1 : \text{The most frequent answer } A_1 \text{ is the true majority.}$$

- Confidence to H_1: Termination based on thresholding Bayes factor:

$$\mathrm{BF}(n) := \frac{\mathbb{P}(\mathcal{D}(n)|H_1)}{\mathbb{P}(\mathcal{D}(n)|H_0)} = \frac{\mathbb{P}(H_1|\mathcal{D}(n))}{\mathbb{P}(H_0|\mathcal{D}(n))} \cdot \frac{\mathbb{P}(H_0)}{\mathbb{P}(H_1)} \quad \text{(Bayes' theorem)}$$
$$\approx s(n) \frac{\mathbb{P}(H_1|\mathcal{D}(n))}{\mathbb{P}(H_0|\mathcal{D}(n))}$$

# of unique answers so far ($s(n)$)

Posterior ratio, modelled as a Dirichlet process

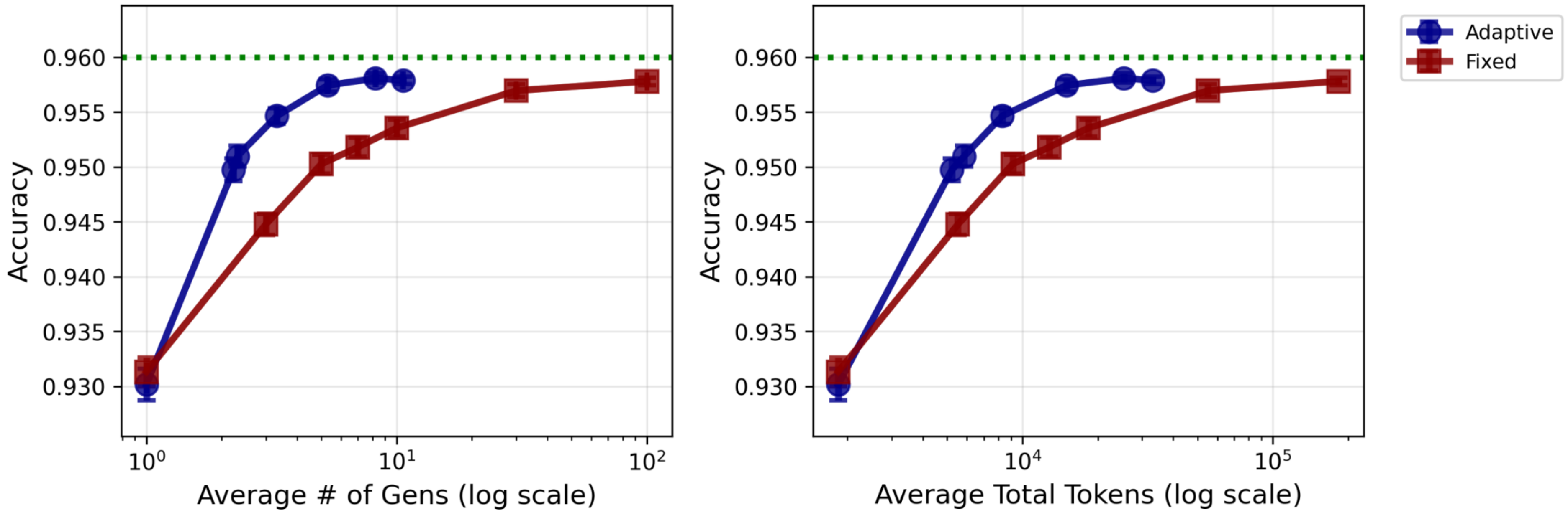
Algorithm for Adaptive Sampling

- Algorithm - Repeat the follows:
  - Sample (= ask LLM to generate) answers from the LLM.
  - Update counts and the Bayes factor after every generation.
  - If $BF \geq B$ (threshold) or had $N_{\max}$ generations, terminate.
- Output the majority answer among collected generations.

- Theorem (consistency): As $B, N_{\max} \to \infty$, the procedure matches Best-of-∞ accuracy almost surely.

Adaptive Sampling: Empirical Performance on MATH500

- Adaptive sampling based on Bayes factor is very efficient:
  - Average $N = 3$ -> comparable to fixed sampling with $N = 10$
  - Average $N = 10$ -> comparable to fixed sampling with $N = 100$



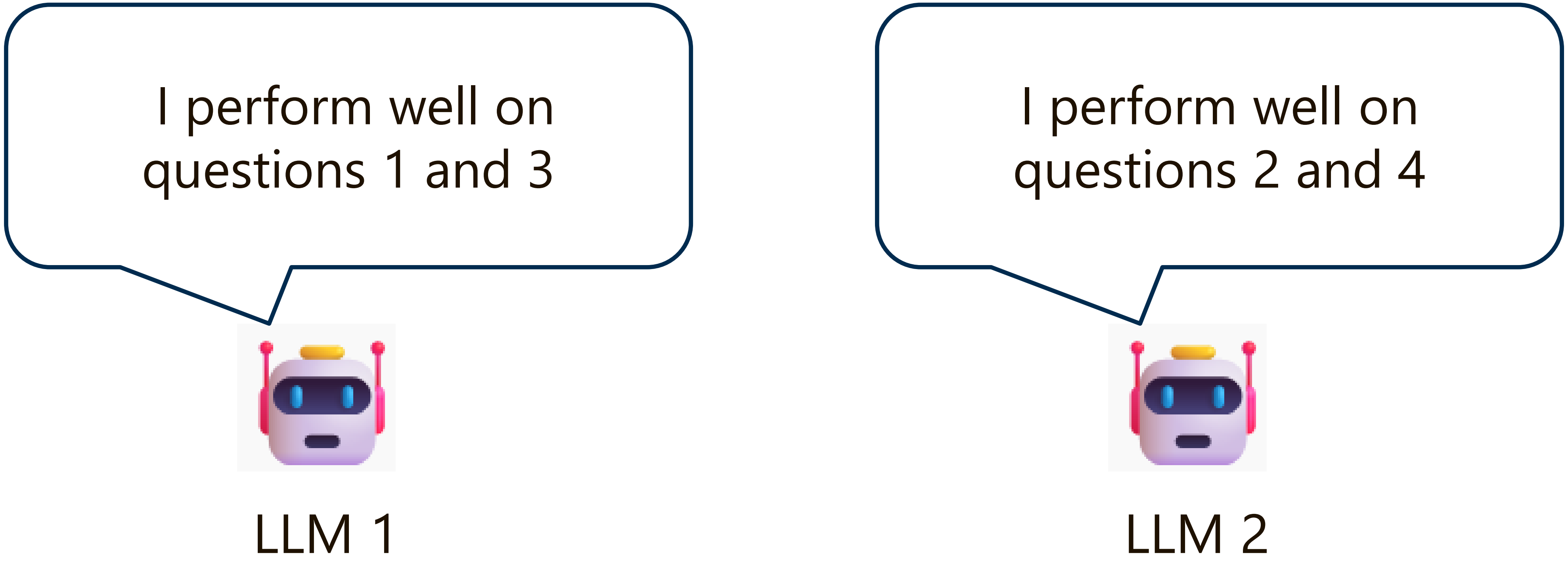
LLM: GPT-OSS-20B, Dataset: MATH500

Agenda

- Introduction
  - Best-of-∞
- Adaptive Sampling:
  - Efficient test-time compute, approaching Best-of-∞ with limited computational budget
- **LLM Ensemble**
  - **Complementary strength of multiple LLMs**
- Summary

Why Ensembles?

- Different LLMs offer complementary strengths on reasoning tasks.
- Weighted sampling per generation lets weaker-but-diverse models contribute.
- Objective: maximize majority-vote accuracy in the limit by **weighting** LLMs.

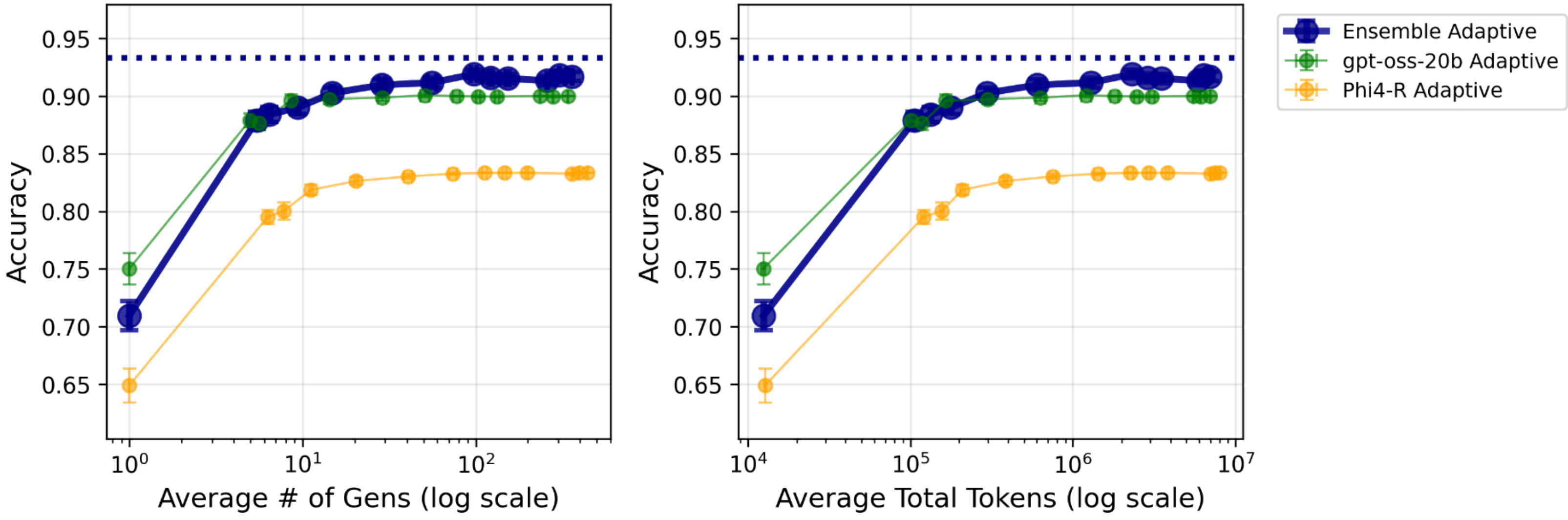


Adaptive LLM Sampling: Algorithm

- Input: List of LLMs $i = 1,2,\ldots,K$
- Repeat the followings:
  - **Select LLM with weight vector $\mathbf{w} = (w_1, w_2, \ldots, w_K)$.**
  - Ask the LLM to generate answers from the LLM.
  - Update counts and the Bayes factor after every generation.
  - If $BF \geq B$ or had $N_{\max}$ generations, terminate.
- Output the majority answer among collected generations.

Best-of-∞ of Ensemble > max Best-of-∞ of single LLM

- Example: On AIME2025,
  - Best-of-∞ of gpt-oss-20b: 0.900 (27/30)
  - Best-of-∞ of Phi-4-reasoning: 0.833 (25/30)
  - Best-of-∞ of weighted mixture: 0.933 (28/30)



Optimal weight

- The weight $w_1, w_2, \dots, w_K$ optimization is reduced to the following **mixed integer linear programming** (MILP).

$$\max_{w \in \Delta^K, y \in \{0,1\}^N} \sum_q y_q$$

$$\text{s.t.} \quad w_i \geq 0 \; \forall_i$$

$$\sum_i w_i = 1$$

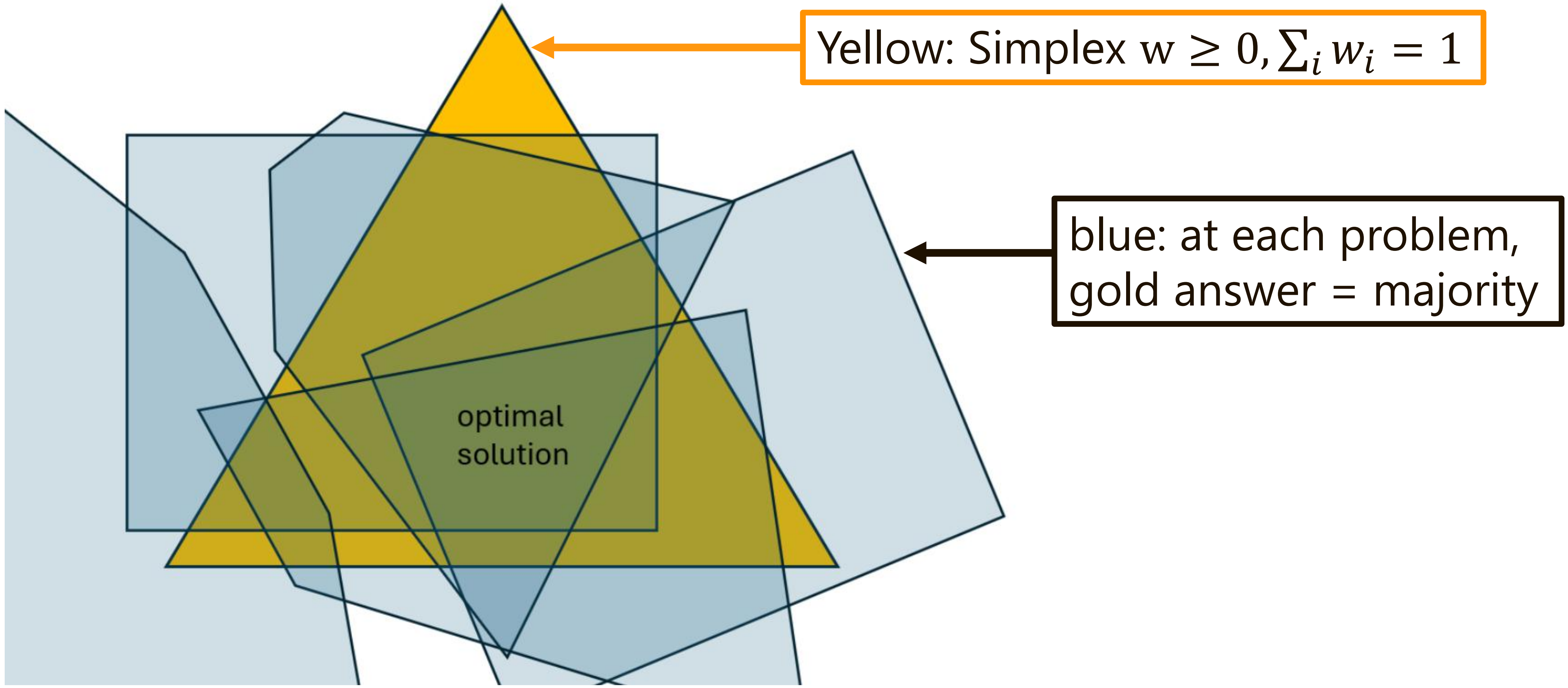
$$A_q w \geq -m(1 - y_q) \; \forall q$$

\# of correctly answered questions

weight is on simplex (summation 1)

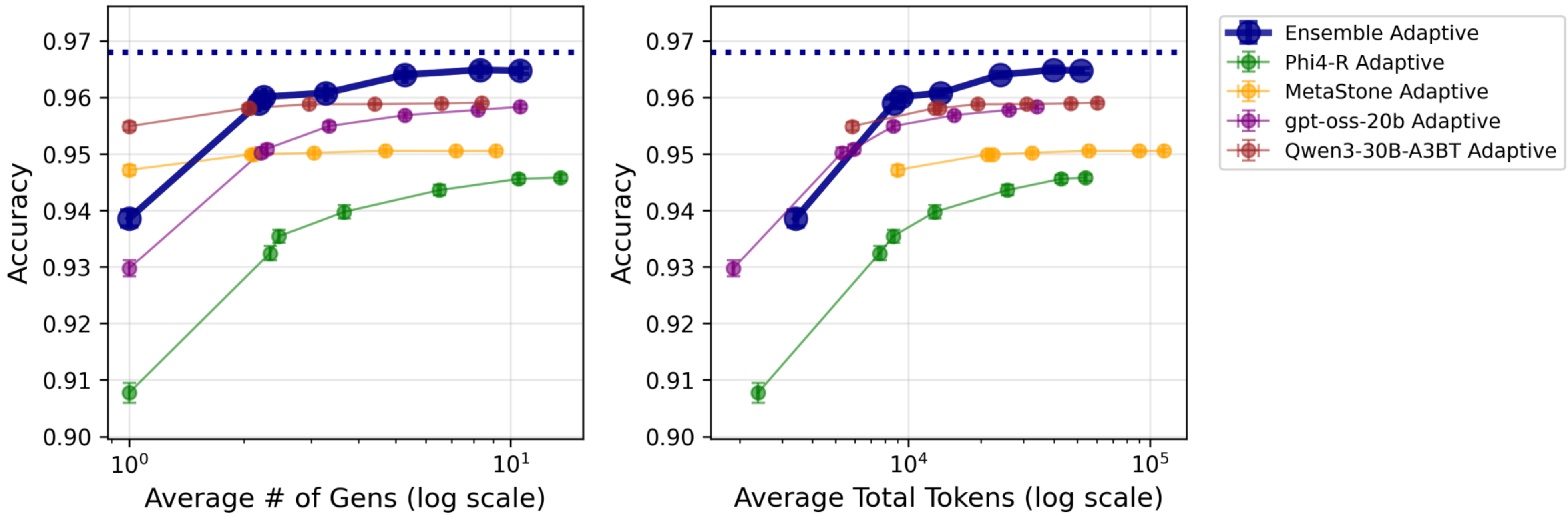
gold answer = population majority

Illustration of the optimal weight



Scalability

- MILP is NP-complete to solve, does not scale for extremely large instance.
- In practice, Optimization easily scales up to ~10 LLMs and ~500 questions.



LLM Ensemble performance in the MATH500 dataset

We have released the generation dataset

- To estimate Best-of-∞ accuracy, we generate ≥ 80 answers for each question.
- Over 500k generated solutions, up to 800M tokens per LLM.

11 LLMs →

LLM	# of files	total generated tokens	total file size (MB)
AM-Thinking-v1	4,800	79,438,111	185.95
Datarus-R1-14B-preview	4,800	49,968,613	127.03
EXAONE-Deep-32B	60,640	478,575,594	1,372.35
GPT-OSS-20B	68,605	244,985,253	98.59[6]
LIMO-v2	6,095	77,460,567	219.45
MetaStone-S1-32B	60,757	806,737,009	2,458.48
NVIDIA-Nemotron-Nano-9B-v2	60,640	295,466,626	897.82
Phi-4-reasoning	168,138	558,980,037	1,841.06
Qwen3-4B	20,640	547,170,887	1,704.28
Qwen3-14B	44,800	666,466,780	1,822.13
Qwen3-30B-A3B-Thinking-2507	60,640	436,865,220	1,234.28

Table 1: Statistics of the large-scale generation dataset that we used in our experiments. Each file corresponds to a single answer. We release it with our code.

Summary

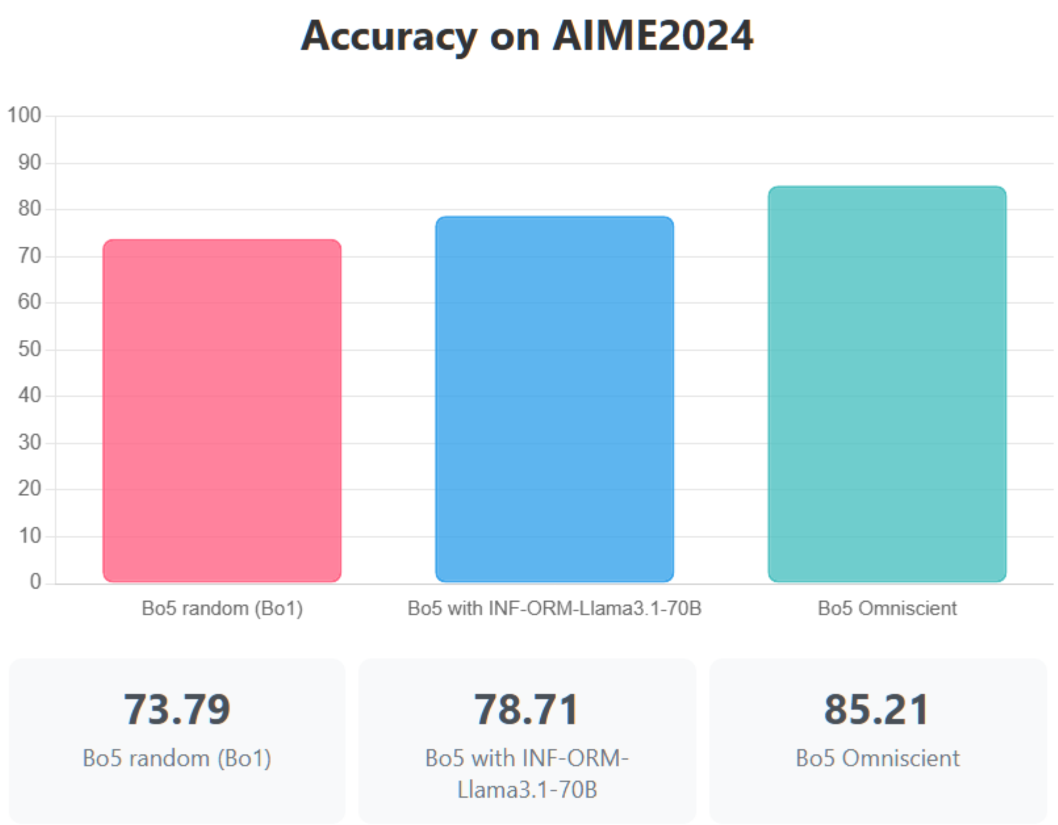
- Best-of-N with adaptive N approaches to Best-of-∞ performance.
- Weighted LLM Ensemble.
  - Outperforms any single LLM.
  - Optimized weights via mixed-integer linear programming (MILP).
- Publicly available dataset: 11 open-weight reasoning LLMs × 4 hard benchmarks × 80+ generations each.

Paper URL (QR below):
https://jkomiyama.github.io/bestofinfty/

(the following are supplementary materials)

LLM reasoning: Best-of-N

- Reward-hacking: Best answer in terms of reward is NOT always the correct answer.
- On heavy-reasoning tasks, best reward models are still random-ish.
- How can we get close to "omniscient" BoN?



Completely Random

Reward Model

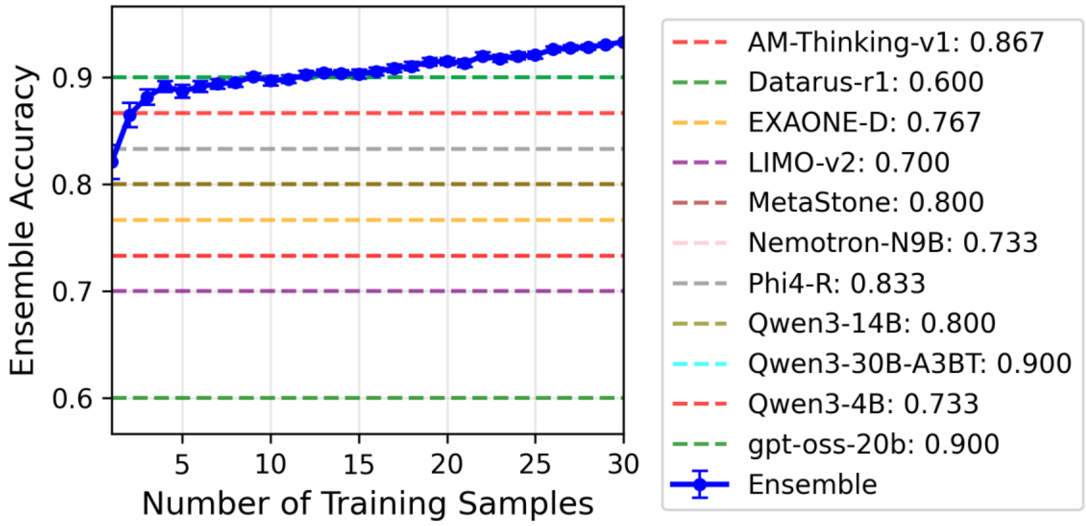
Omniscient (gold answer)

Additional experimental findings

- Transfer learning: Weights trained on AIME2024 beat or tie best single model on AIME2025 in 64% of 165 combinations.
  - Moderately effective?
- Selection comparison (Bo5 on AIME2025): Majority voting outperforms reward models and LLM-as-a-judge variants.
  - Majority voting is still one of the most useful aggregation.

How many gold answers are needed to obtain good weights?

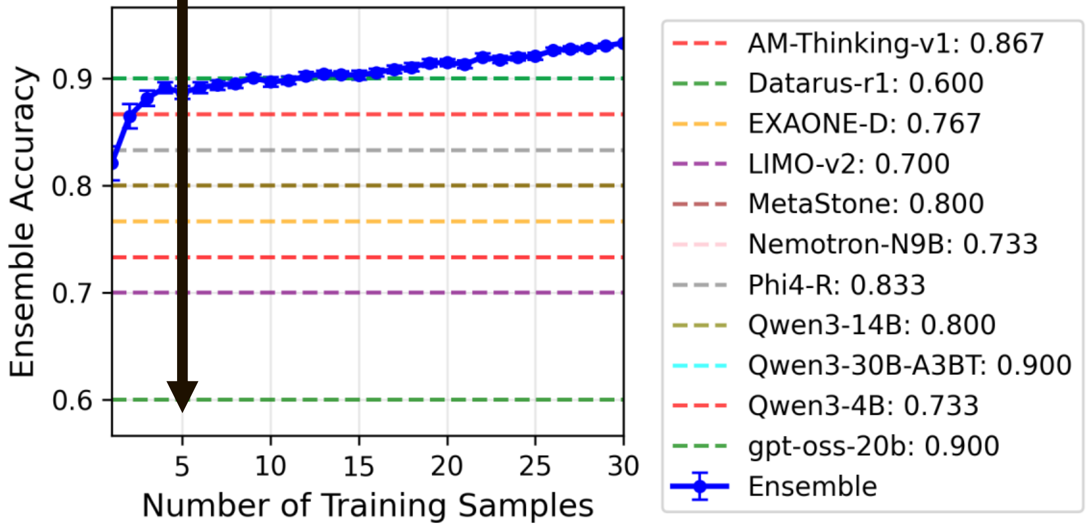
AIME2025,
11 LLMs



How many gold answers are needed to obtain good weights?

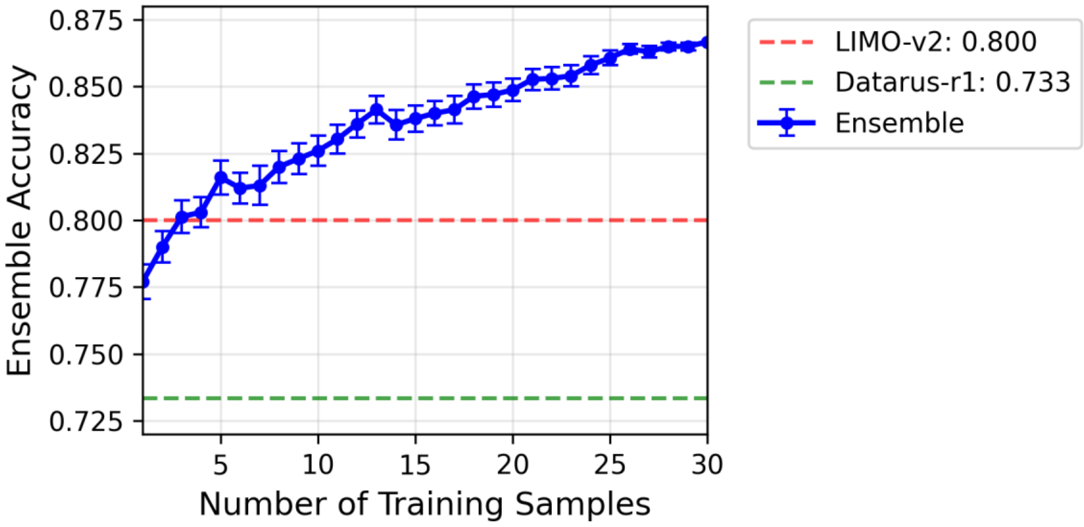
5 gold answers to find a reasonable weight vector

AIME2025,
11 LLMs



How many gold answers are needed to obtain good weights?

AIME2024,
2 LLMs



AIME2025,
11 LLMs