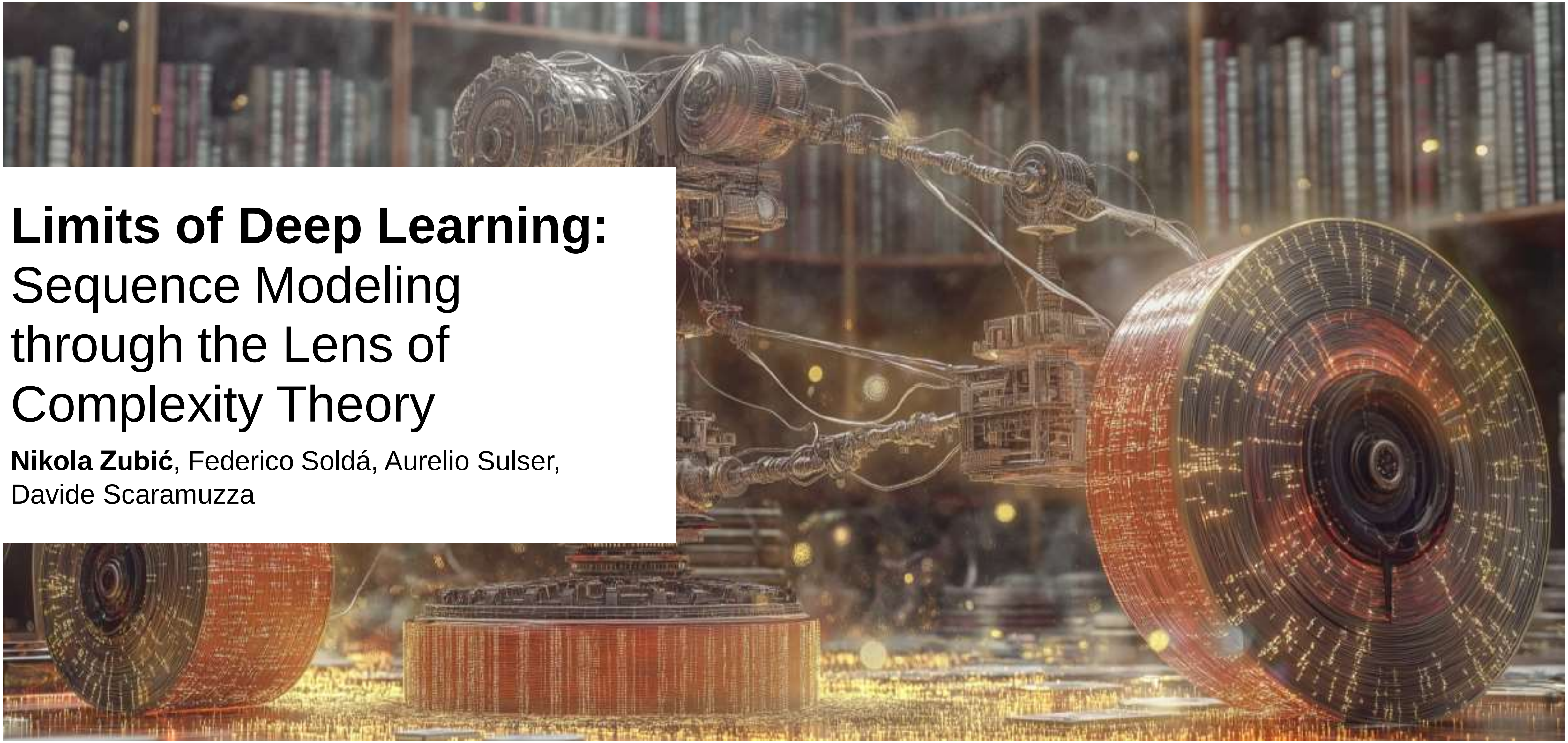
Universität Zürich UZH

ETH zürich

ROBOTICS & PERCEPTION GROUP



# **Limits of Deep Learning:** Sequence Modeling through the Lens of Complexity Theory

**Nikola Zubić**, Federico Soldá, Aurelio Sulser,
Davide Scaramuzza

# Why Study Limits of Deep Learning Models?

- **Successes of Deep Learning**: Transformers, LLMs, SSMs excel at language, vision, etc.
- **The Gap**: Persistent failures in multi-step reasoning, compositional tasks, and function composition
    - Hallucinations
- **Our Focus**: Theoretical lens + empirical validation to identify where and why these models fall short

RoboCat: A self-improving robotic agent

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, Yi Zhang

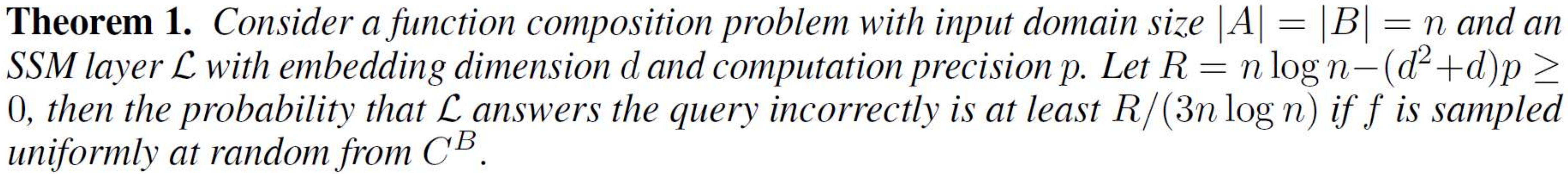
Artificial intelligence (AI) researchers have been developing and refining large language models (LLMs) that exhibit remarkable capabilities across a variety of domains and tasks, challenging our understanding of learning and cognition. The latest model developed by OpenAI, GPT-4, was trained using an unprecedented scale of compute and data. In this paper, we report on our investigation of an early version of GPT-4, when it was still in active development by OpenAI. We contend that (this early version of) GPT-4 is part of a new cohort of LLMs (along with ChatGPT and Google's PaLM for example) that exhibit more general intelligence than previous AI models. We discuss the rising capabilities and implications of these models. We demonstrate that, beyond its mastery of language, GPT-4 can solve novel and difficult tasks that span mathematics, coding, vision, medicine, law, psychology and more, without needing any special prompting. Moreover, in all of these tasks, GPT-4's performance is strikingly close to human-level performance, and often vastly surpasses prior models such as ChatGPT. Given the breadth and depth of GPT-4's capabilities, we believe that it could reasonably be viewed as an early (yet still incomplete) version of an artificial general intelligence (AGI) system. In our exploration of GPT-4, we put special emphasis on discovering its limitations, and we discuss the challenges ahead for advancing towards deeper and more comprehensive versions of AGI, including the possible need for pursuing a new paradigm that moves beyond next-word prediction. We conclude with reflections on societal influences of the recent technological leap and future research directions.

OpenAI o1; Google Gemini 1.5 Pro; [Bousmalis , 2023] RoboCat: A self-improving robotic agent; [Bubeck, 2023] Sparks of Artificial General Intelligence: Early experiments with GPT-4

# Related Work

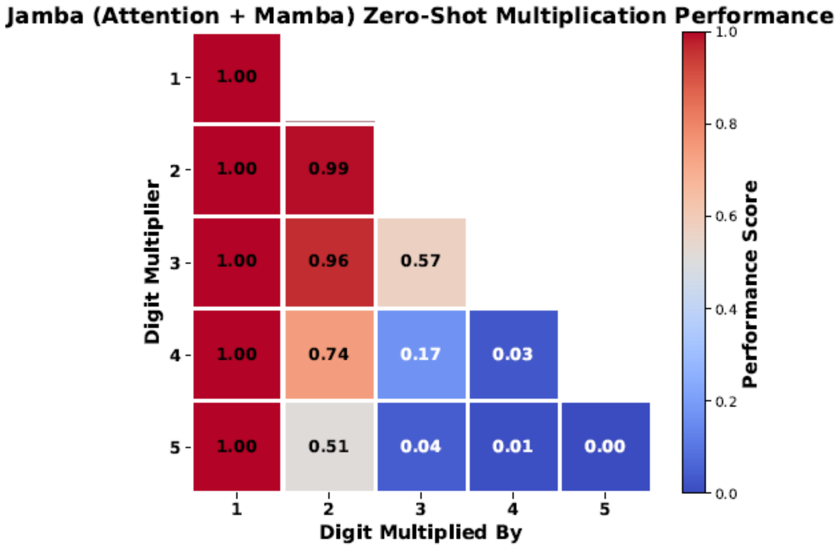
1. **Empirical Limitations in Function Composition and Reasoning**
    - **Key Idea**: Transformers struggle with deeply compositional tasks (Dziri et al., 2023).
    - **Why It Matters**: Core applications (e.g., math, logical reasoning) demand multi-step function composition; current architectures fall short.
2. **Chain-of-Thought Prompting**
    - **Key Idea**: CoT helps break down complex tasks (Wei et al., 2022).
    - **Interesting question**: But is it sufficient for inherently complex tasks? Can it fully solve the problem?
3. **Expressive Power and Complexity**
    - **Key Idea**: Neural networks have constraints; Transformers are in logspace-uniform $TC^0$ (Merrill et al., 2024).
    - **Why It Matters**: This aligns with formal language theory—finite precision restricts them to regular languages or log-space computations.
4. **Alternative Approaches**
    - **Key Idea**: External memory modules, symbolic components, or neuro-symbolic systems (Graves et al., 2016; Dai et al., 2019) can mitigate some issues.
    - **Why It Matters**: Overcoming SSM/Transformer limits may require going beyond standard deep nets—e.g., memory-augmented or hybrid models.

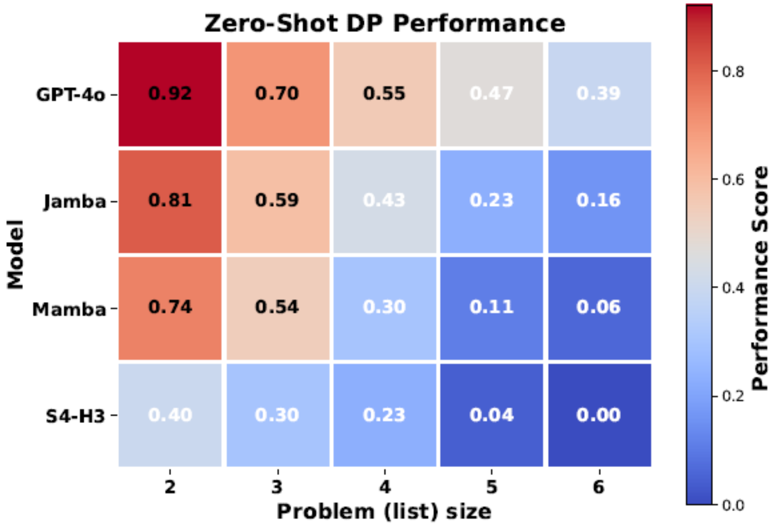
# Function Composition Over Large Domains

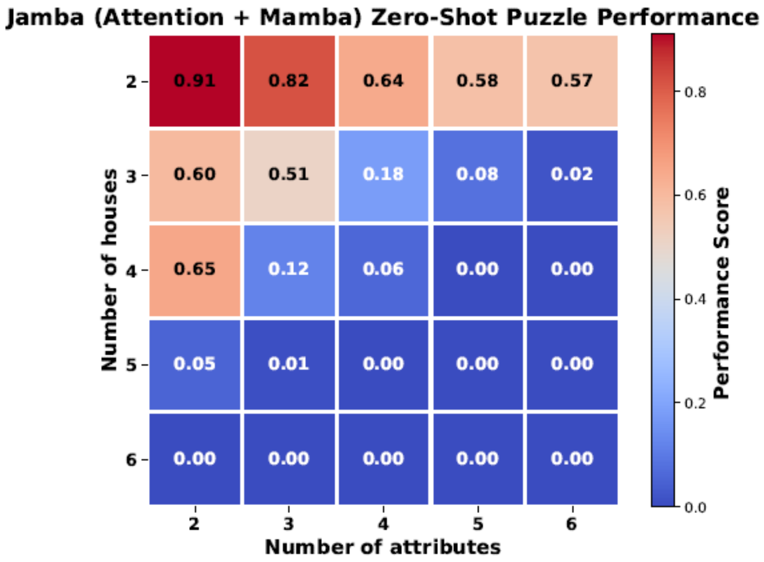


**Theorem 1.** *Consider a function composition problem with input domain size $|A| = |B| = n$ and an SSM layer $\mathcal{L}$ with embedding dimension $d$ and computation precision $p$. Let $R = n \log n - (d^2 + d)p \geq 0$, then the probability that $\mathcal{L}$ answers the query incorrectly is at least $R/(3n \log n)$ if $f$ is sampled uniformly at random from $C^B$.*

- **Statement:** One-layer SSMs cannot solve function composition on large domains without an impractically large state.
- **Implication:** To get high accuracy, the state dimension or precision must grow at least on the order of $n\log n$. Otherwise, the model errs with high probability.

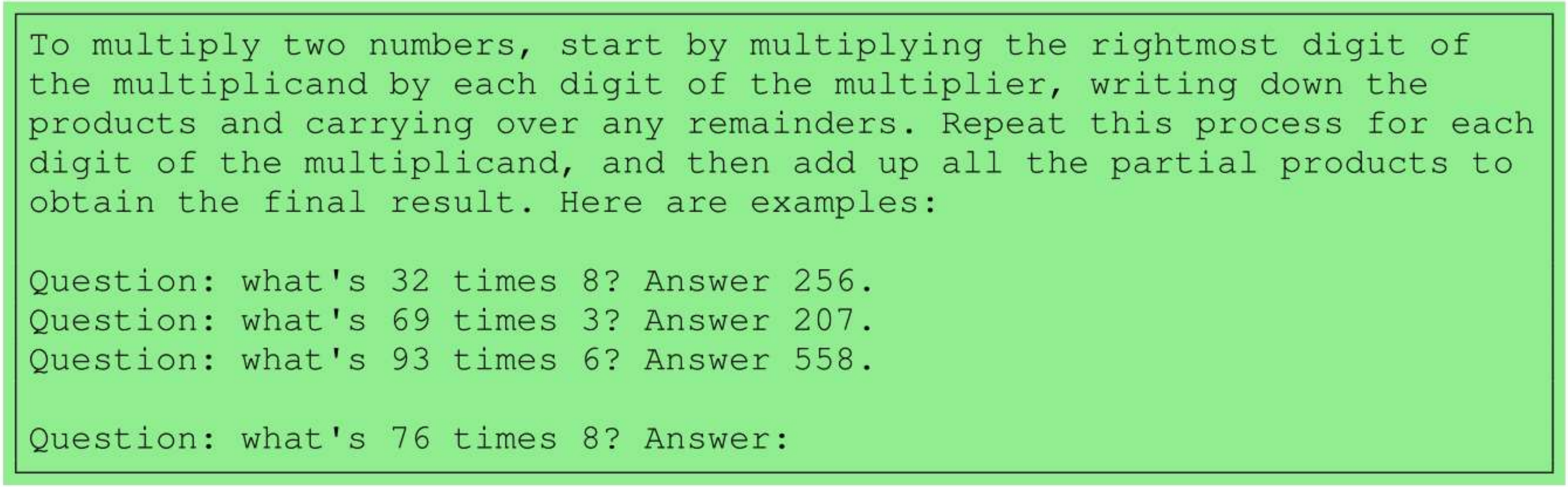






# Understanding Chain-of-Thought (CoT) in Language Models

- Definition of CoT:
  - —A series of intermediate natural language reasoning steps leading to the final output
- Purpose of CoT: Mitigates hallucinations by breaking down tasks into manageable steps
  - —Wei et al., NIPS 2022: Demonstrated CoT's effectiveness in reducing errors

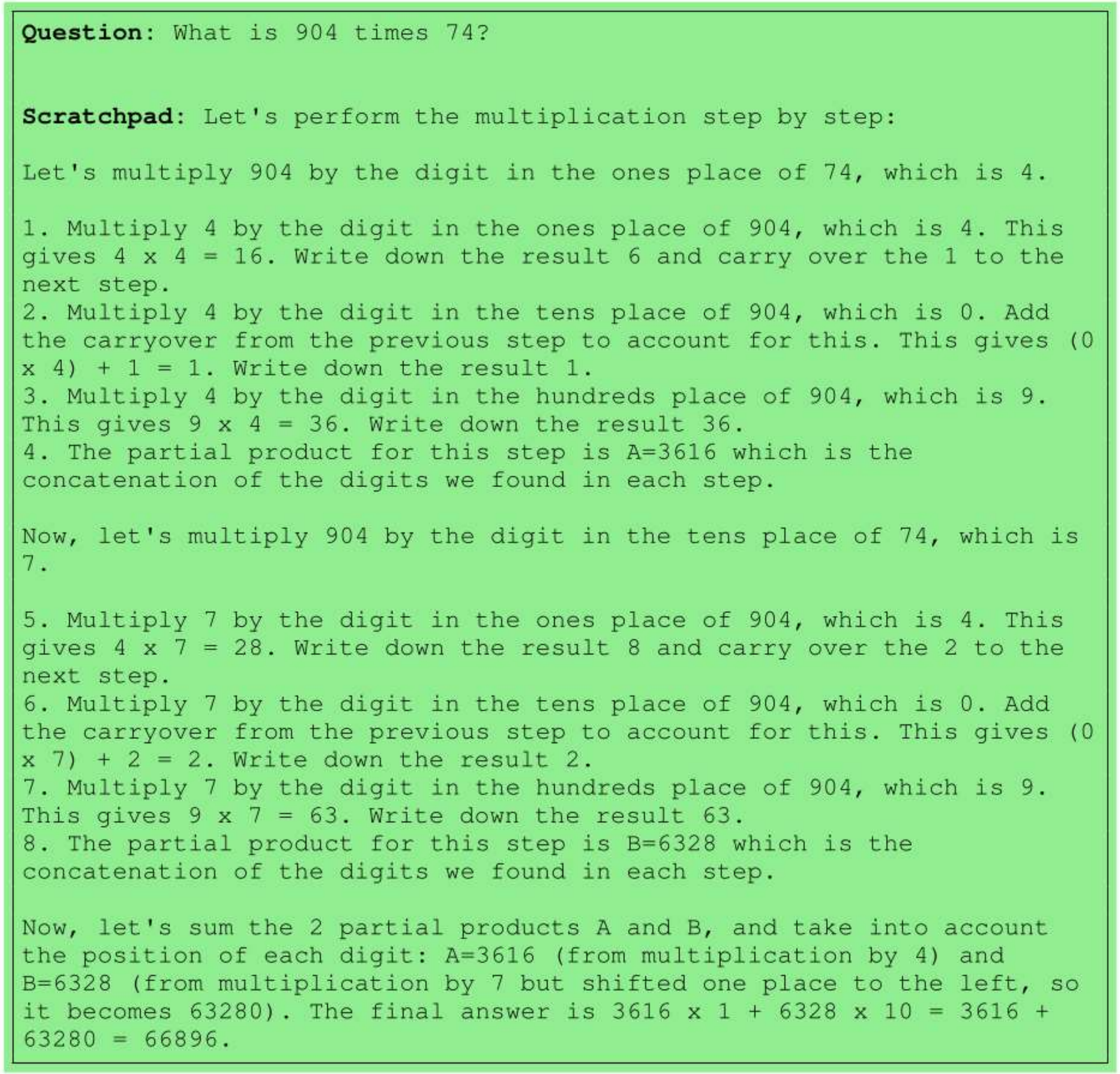


To multiply two numbers, start by multiplying the rightmost digit of the multiplicand by each digit of the multiplier, writing down the products and carrying over any remainders. Repeat this process for each digit of the multiplicand, and then add up all the partial products to obtain the final result. Here are examples:

Question: what's 32 times 8? Answer 256.
Question: what's 69 times 3? Answer 207.
Question: what's 93 times 6? Answer 558.

Question: what's 76 times 8? Answer:

# Compositional tasks – multi-digit multiplication



**Question:** What is 904 times 74?

**Scratchpad:** Let's perform the multiplication step by step:

Let's multiply 904 by the digit in the ones place of 74, which is 4.

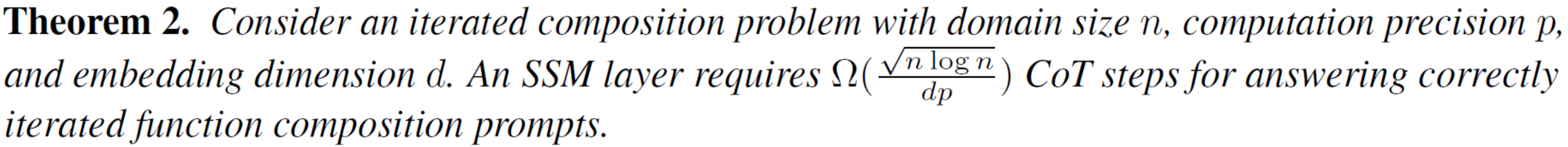
1. Multiply 4 by the digit in the ones place of 904, which is 4. This gives 4 x 4 = 16. Write down the result 6 and carry over the 1 to the next step.
2. Multiply 4 by the digit in the tens place of 904, which is 0. Add the carryover from the previous step to account for this. This gives (0 x 4) + 1 = 1. Write down the result 1.
3. Multiply 4 by the digit in the hundreds place of 904, which is 9. This gives 9 x 4 = 36. Write down the result 36.
4. The partial product for this step is A=3616 which is the concatenation of the digits we found in each step.

Now, let's multiply 904 by the digit in the tens place of 74, which is 7.

5. Multiply 7 by the digit in the ones place of 904, which is 4. This gives 4 x 7 = 28. Write down the result 8 and carry over the 2 to the next step.
6. Multiply 7 by the digit in the tens place of 904, which is 0. Add the carryover from the previous step to account for this. This gives (0 x 7) + 2 = 2. Write down the result 2.
7. Multiply 7 by the digit in the hundreds place of 904, which is 9. This gives 9 x 7 = 63. Write down the result 63.
8. The partial product for this step is B=6328 which is the concatenation of the digits we found in each step.

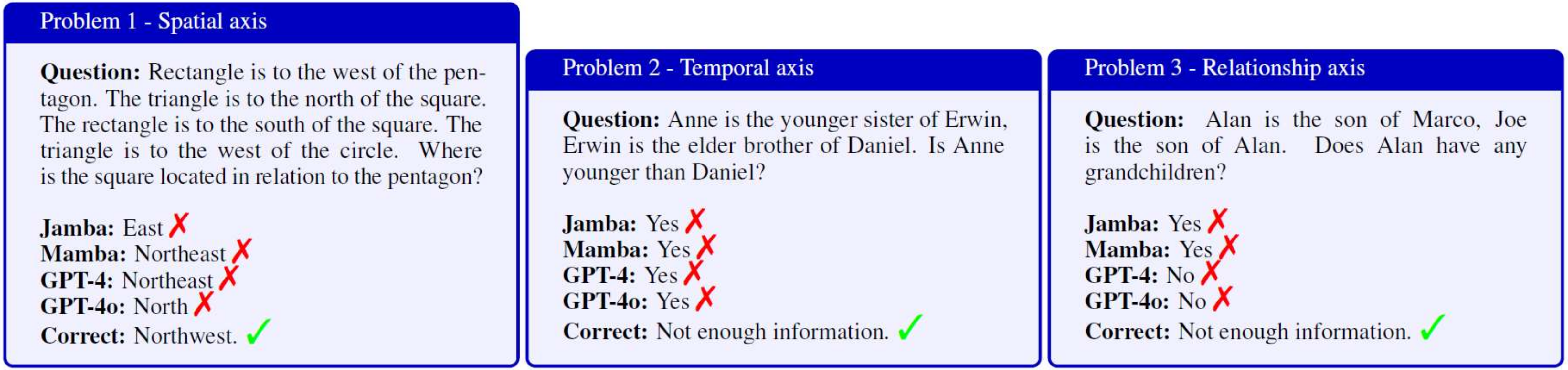
Now, let's sum the 2 partial products A and B, and take into account the position of each digit: A=3616 (from multiplication by 4) and B=6328 (from multiplication by 7 but shifted one place to the left, so it becomes 63280). The final answer is 3616 x 1 + 6328 x 10 = 3616 + 63280 = 66896.

# Chain-of-Thought Steps in Iterated Composition



**Theorem 2.** *Consider an iterated composition problem with domain size $n$, computation precision $p$, and embedding dimension $d$. An SSM layer requires $\Omega(\frac{\sqrt{n \log n}}{dp})$ CoT steps for answering correctly iterated function composition prompts.*

- **Statement:** Even with CoT prompting, one-layer SSMs need a polynomially growing number of steps to solve iterated function composition
- **Implication:** Merely providing multiple reasoning steps (like scratchpads) does not fundamentally fix the capacity issue; it only trades off more (potentially many more) tokens to handle complex compositions.



**Problem 1 - Spatial axis**

**Question:** Rectangle is to the west of the pentagon. The triangle is to the north of the square. The rectangle is to the south of the square. The triangle is to the west of the circle. Where is the square located in relation to the pentagon?

**Jamba:** East ✗
**Mamba:** Northeast ✗
**GPT-4:** Northeast ✗
**GPT-4o:** North ✗
**Correct:** Northwest. ✓

**Problem 2 - Temporal axis**

**Question:** Anne is the younger sister of Erwin, Erwin is the elder brother of Daniel. Is Anne younger than Daniel?

**Jamba:** Yes ✗
**Mamba:** Yes ✗
**GPT-4:** Yes ✗
**GPT-4o:** Yes ✗
**Correct:** Not enough information. ✓

**Problem 3 - Relationship axis**

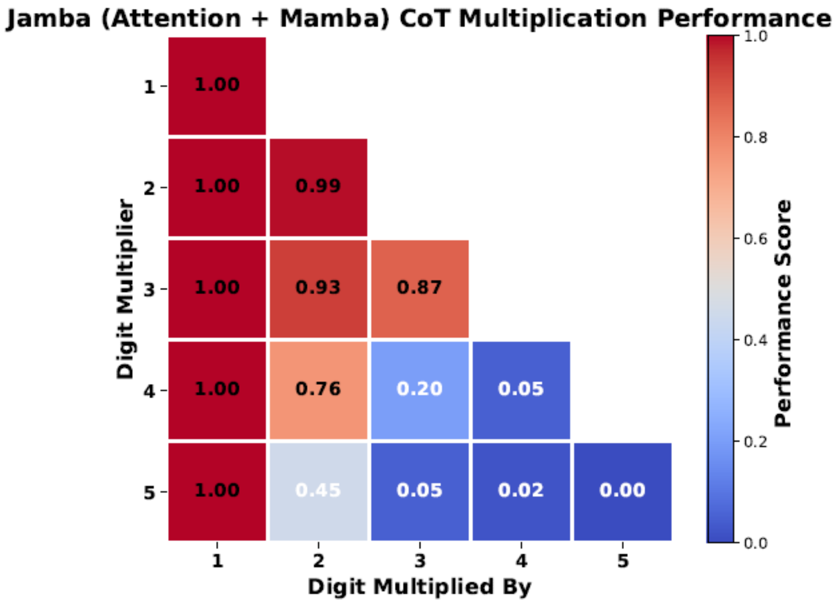
**Question:** Alan is the son of Marco, Joe is the son of Alan. Does Alan have any grandchildren?

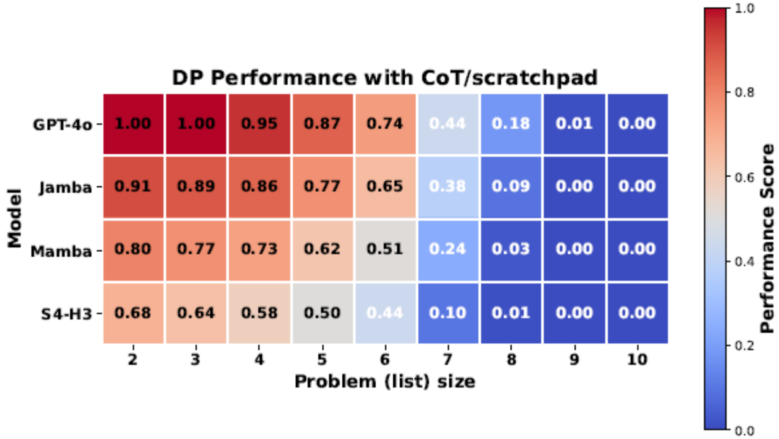
**Jamba:** Yes ✗
**Mamba:** Yes ✗
**GPT-4:** No ✗
**GPT-4o:** No ✗
**Correct:** Not enough information. ✓

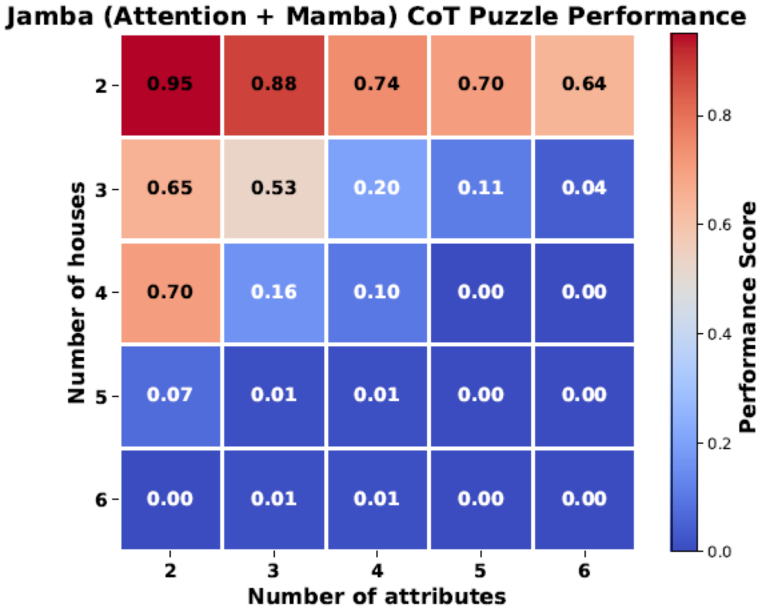
# Chain-of-Thought Steps in Iterated Composition

**Theorem 2.** *Consider an iterated composition problem with domain size $n$, computation precision $p$, and embedding dimension $d$. An SSM layer requires $\Omega(\frac{\sqrt{n \log n}}{dp})$ CoT steps for answering correctly iterated function composition prompts.*

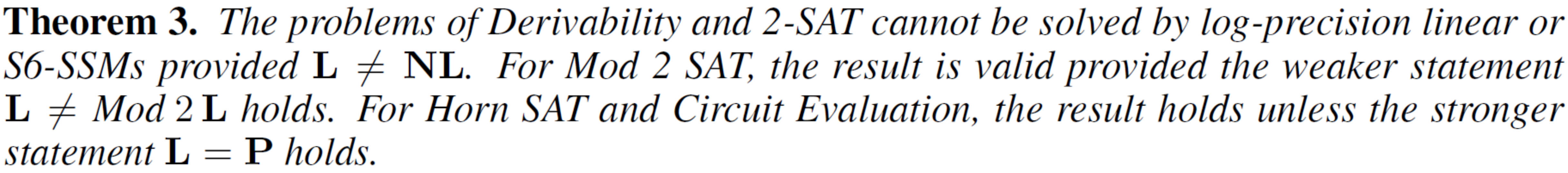
- **Statement:** Even with CoT prompting, one-layer SSMs need a polynomially growing number of steps to solve iterated function composition
- **Implication:** Merely providing multiple reasoning steps (like scratchpads) does not fundamentally fix the capacity issue; it only trades off more (potentially many more) tokens to handle complex compositions.



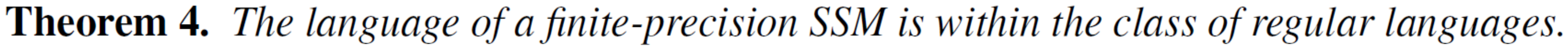




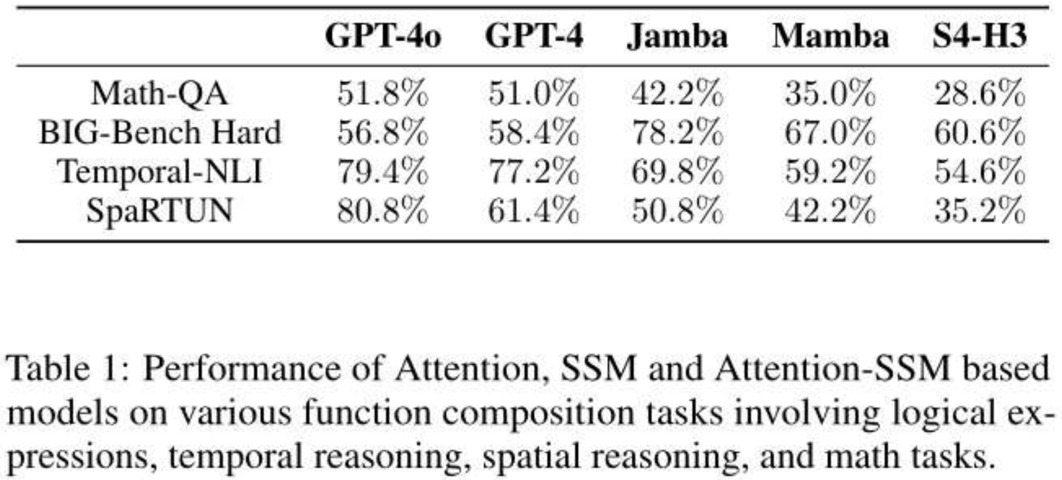
# SSMs Are in $L$; Consequence for $NL$-Complete Problems
# Regular Language Limitation Under Finite Precision



**Theorem 3.** *The problems of Derivability and 2-SAT cannot be solved by log-precision linear or S6-SSMs provided* $\mathbf{L} \neq \mathbf{NL}$. *For Mod 2 SAT, the result is valid provided the weaker statement* $\mathbf{L} \neq \mathit{Mod}\, 2\, \mathbf{L}$ *holds. For Horn SAT and Circuit Evaluation, the result holds unless the stronger statement* $\mathbf{L} = \mathbf{P}$ *holds.*

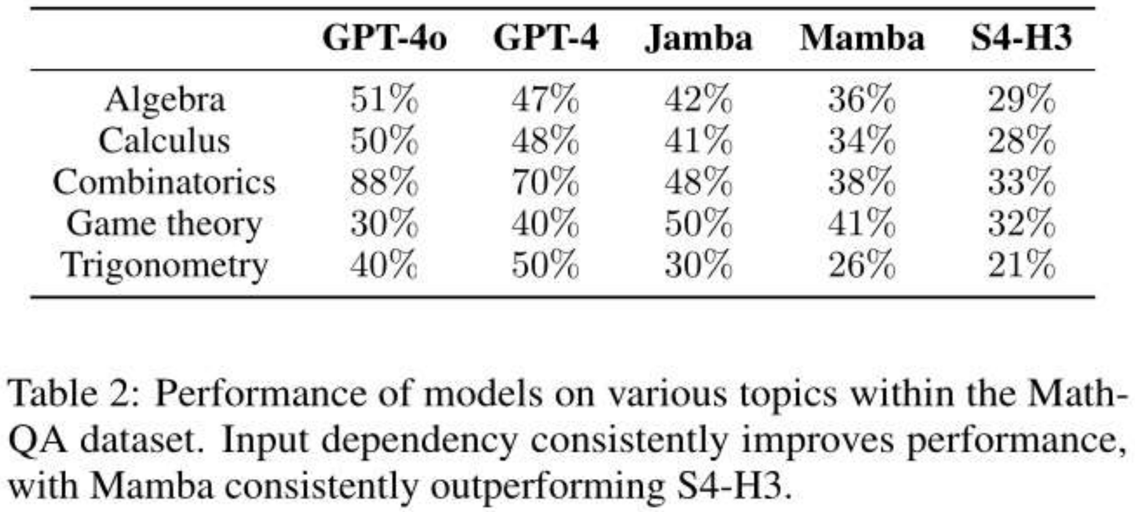


**Theorem 4.** *The language of a finite-precision SSM is within the class of regular languages.*



| | GPT-4o | GPT-4 | Jamba | Mamba | S4-H3 |
|---|---|---|---|---|---|
| Math-QA | 51.8% | 51.0% | 42.2% | 35.0% | 28.6% |
| BIG-Bench Hard | 56.8% | 58.4% | 78.2% | 67.0% | 60.6% |
| Temporal-NLI | 79.4% | 77.2% | 69.8% | 59.2% | 54.6% |
| SpaRTUN | 80.8% | 61.4% | 50.8% | 42.2% | 35.2% |

Table 1: Performance of Attention, SSM and Attention-SSM based models on various function composition tasks involving logical expressions, temporal reasoning, spatial reasoning, and math tasks.



| | GPT-4o | GPT-4 | Jamba | Mamba | S4-H3 |
|---|---|---|---|---|---|
| Algebra | 51% | 47% | 42% | 36% | 29% |
| Calculus | 50% | 48% | 41% | 34% | 28% |
| Combinatorics | 88% | 70% | 48% | 38% | 33% |
| Game theory | 30% | 40% | 50% | 41% | 32% |
| Trigonometry | 40% | 50% | 30% | 26% | 21% |

Table 2: Performance of models on various topics within the Math-QA dataset. Input dependency consistently improves performance, with Mamba consistently outperforming S4-H3.

# Conclusions and Future Directions

—**Fundamental Barriers**: Current architectures—finite-precision SSMs, Transformers—are inherently limited in multi-step reasoning.

—**Need for Novel Architectures**: Symbolic components? Larger or external memory? Hybrid neuro-symbolic approaches?

—**Toward General Intelligence**: Must tackle these computational and representational constraints to handle true compositional reasoning.

**Conference Poster Session**
**Friday, April 25 • 10:00 a.m. – 12:30 p.m. (GMT+8)**

Workshops:

- Reasoning and Planning for Large Language Models
- New Frontiers in Associative Memories
- XAI4Science: From Understanding Model Behavior to Discovering New Scientific Knowledge