One Bit at a Time: Universal Per-Step Error Rates in LLM Compositional Reasoning

Anonymous Authors

Abstract

We report a universal law governing how large language models fail at compositional reasoning. On a controlled finite-state machine tracing task parameterized by state complexity k (bits per step) and trace length n (number of steps), we find that chain-of-thought accuracy follows $\eta = k \cdot \exp(-a \cdot k \cdot n)$, where a is a single per-step-per-bit error rate characteristic of each model. This formula, derived from the assumption that each reasoning step independently corrupts each bit of carried state, fits 9 out of 10 tested models with $R^2 > 0.7$ (8 with $R^2 > 0.8$). The parameter a serves as a **model signature**: it ranks models in exact agreement with their perceived quality tiers (Claude Sonnet 4: $a = 0.00010$; GPT-4o-mini: $a = 0.00705$; Qwen 7B: $a = 0.06574$). Three findings emerge. First, the formula fits only chain-of-thought traces, not baseline (direct-answer) inference—every baseline model yields negative R^2, indicating that baseline reasoning operates under a qualitatively different error regime where per-step independence breaks down. Second, the formula implies a hard capacity bound: reliable compositional reasoning requires $k \cdot n < 1/a$, giving each model a quantitative "reasoning horizon." Third, two controlled experiments—token-count-matched ablation and gold-intermediate decomposition—confirm that the compositional bottleneck is working memory (carrying intermediate state), not retrieval (reading from context). These results provide a simple, predictive theory of compositional reasoning failure: every model has a characteristic error rate, and that rate determines exactly when and how multi-step reasoning breaks down.

1 Introduction

When a language model fails at a multi-step reasoning task, we lack a quantitative framework for explaining *why*. Was the task too complex for a single step? Did errors accumulate across steps? Would a better model help, or would more chain-of-thought tokens suffice? Current evaluations treat reasoning benchmarks as monolithic scores, obscuring the mechanistic structure of failure.

We propose that compositional reasoning failure in LLMs follows a remarkably simple law. Consider a task requiring the model to carry k bits of state through n sequential steps, using chain-of-thought. If each step independently corrupts each carried bit with probability $\epsilon = a \cdot k$, the probability of arriving at the correct final state is:

$$P(\text{correct}) = (1 - ak)^n \approx \exp(-akn) \tag{1}$$

The expected bits correctly resolved are then:

$$\eta = k \cdot \exp(-akn) \tag{2}$$

where a is a single scalar—the **per-step-per-bit error rate**—that characterizes the model. This is the entire theory. One parameter per model, fit from data, explains accuracy across all combinations of k and n.

This paper tests Equation 2 on 10 language models spanning five provider families and nearly three orders of magnitude in the error rate a. We find that a ranks models with striking fidelity: frontier models achieve $a \sim 10^{-4}$ (one error per ten thousand step-bits), while small open-weight models reach $a \sim 10^{-2}$ (one error per hundred step-bits). The formula fits chain-of-thought data with high R^2 but fails completely on baseline (no-CoT) inference, revealing a qualitative difference in how errors propagate with and without externalized state.

Contributions. (1) We derive a one-parameter model of compositional reasoning accuracy from per-step error independence (§3). (2) We validate it on 10 models across 5 provider families with a controlled FSM tracing task, showing universal fit under CoT and universal failure under baseline (§5). (3) We establish a as a model signature with bootstrap 95% CIs that ranks models consistently with perceived quality (§5.2). (4) We prove via controlled experiments that the bottleneck is working memory, not retrieval, and that for weak models the primary CoT benefit is compute budget rather than explicit state externalization (§5.4–**??**). (5) We test on two additional tasks (XOR chains, modular arithmetic) and show the formula applies specifically to unstructured state tracking (§5.6). (6) We independently validate the exponential form via Fano's inequality ($a_{\text{IT}}/a_{\text{fit}} \approx 2$, $r = 0.99$ across 3 orders of magnitude) and out-of-sample prediction.

2 Related Work

Chain-of-thought reasoning. Wei et al. Wei et al. [2022] demonstrated that chain-of-thought prompting dramatically improves multi-step reasoning. Subsequent work explored variants—tree-of-thought Yao et al. [2023], self-consistency Wang et al. [2023], process reward models Lightman et al. [2024]—but the mechanistic question remains: *why* does externalizing intermediate steps help? Our framework provides a precise answer: CoT converts the error regime from entangled (where per-step independence breaks down) to independent (where errors accumulate multiplicatively and predictably).

Transformer expressiveness. Merrill and Sabharwal Merrill and Sabharwal [2024] proved that transformers with $O(n)$ CoT steps can recognize all regular languages, while Feng et al. Feng et al. [2024] showed that without CoT, transformers are limited to AC^0. These results characterize *which* problems are solvable in principle. Our work addresses the complementary question: for problems within the solvable class, *how well* does a specific model perform as task parameters vary?

Working memory in LLMs. Gong et al. Gong et al. [2024] tested LLMs on n-back tasks and found working memory limits correlated with reasoning failure. Li et al. Li et al. [2025] analyzed factors constraining LLM working memory. These are empirical observations without a predictive formula. Our Equation 2 provides one: given a, k, and n, we predict accuracy before running any experiment.

State tracking. Liu et al. Liu et al. [2023] and Kim and Schuster Kim et al. [2023] study how transformers track state in formal languages, including mechanistic analyses of how attention heads implement finite automata. Our FSM tracing task is designed to isolate pure state tracking from pattern matching, and our contribution is the quantitative error law rather than the mechanistic circuit.

Hardware roofline model. Williams et al. Williams et al. [2009] decomposed hardware performance into compute-bound and memory-bound regimes via arithmetic intensity. The present paper adapts this framework to cognitive performance, showing that a single exponential with one free parameter suffices.

3 Task Design and Theoretical Framework

3.1 Random FSM Tracing

We design a parameterized task that isolates compositional state tracking from all other cognitive demands. Each instance consists of:

1. A random deterministic finite automaton (DFA) with 2^k states and 4 input symbols $\{A, B, C, D\}$.
2. A transition table with 4×2^k entries, each drawn uniformly at random.
3. A random start state and a random input sequence of length n.
4. The model must output the final state after simulating all n transitions.

Anti-memorization. The transition table is freshly randomized for every instance. No two instances share the same automaton. The model cannot exploit distributional regularities—it must perform genuine sequential state tracking.

Parameters. State complexity $k \in \{1, 2, 3, 4, 5, 6\}$ controls the bits of state the model must carry (the state space is $\{S_0, \ldots, S_{2^k-1}\}$). Trace length n ranges from 1 to 40 across experiments. We evaluate 30–50 instances per (k, n) cell with exact-match accuracy.

Chain-of-thought prompting. Under CoT, the model is instructed to show each step as "Step i: $S_x + Y \to S_z$" before giving the final state. Under baseline, the model outputs only the final state label.

3.2 Derivation of the Error Law

Consider a model performing n sequential state transitions, each requiring it to carry k bits of state. We make one assumption:

Per-step-per-bit independence. Each step independently corrupts each bit of carried state with probability p, where $p = a \cdot k$ for a model-specific constant a.

The linear dependence $p = ak$ reflects that more bits of state create more opportunities for error per step—a natural consequence of attention heads sharing bandwidth across state bits. The probability of carrying all k bits correctly through one step is $(1 - ak)^1 \approx 1 - ak$ for small ak. After n steps:

$$P(\text{all bits correct after } n \text{ steps}) = (1 - ak)^n \approx \exp(-akn) \tag{3}$$

The expected bits of state correctly resolved—our performance metric—is:

$$\eta = k \cdot P(\text{correct}) = k \cdot \exp(-akn) \tag{4}$$

This is the complete model. It has one free parameter per model (a) and makes quantitative predictions for all (k, n) combinations.

Dimensionless form. Define the dimensionless product $\Pi = akn$. Then $\eta/k = \exp(-\Pi)$. All models, tasks, and trace lengths should collapse onto a single curve when plotted as normalized accuracy η/k versus Π. This is the universal collapse we test in §5.

Capacity bound. Equation 4 implies that η is non-negligible only when $akn \lesssim 1$, i.e., $k \cdot n < 1/a$. This defines a **reasoning horizon**: the maximum product of state complexity and trace length that a model can handle. For Claude Sonnet 4 ($a = 0.00010$), $kn <$ 10,000; for Qwen 7B ($a = 0.06574$), $kn < 15$.

3.3 Why Per-Step Independence Requires CoT

The derivation assumes that errors at step t are independent of errors at step $t-1$. This requires that the model has access to the *correct* intermediate state at each step. Under chain-of-thought, the model writes the intermediate state to its output buffer, then reads it back as input for the next step. Each step is therefore a fresh table lookup from an explicitly stated starting state—the independence assumption is mechanistically justified.

Under baseline (no CoT), the model must carry state entirely within its hidden representations. Errors at one step corrupt the internal state, which propagates to all subsequent steps. The per-step independence assumption breaks down: errors are correlated, compounding in ways that depend on the model's internal state-tracking mechanism. This is why we expect—and observe—that Equation 4 fits CoT data but not baseline data.

4 Experimental Setup

4.1 Models

We evaluate 10 models spanning five provider families and nearly three orders of magnitude in the error rate a (Table 1).

Table 1: Models evaluated. API models use default parameters. Local models run quantized (Q4_K_M) via Ollama on CPU with temperature 0.

Model	Provider	Access	Params
Claude Sonnet 4	Anthropic	API	undisclosed
Claude Haiku 3.5	Anthropic	API	undisclosed
GPT-4o	OpenAI	API	undisclosed
GPT-4o-mini	OpenAI	API	undisclosed
Qwen2.5-32B-Instruct	Alibaba	local	32B
LLaMA-3.1-70B-Instruct	Meta	local	70B
Gemma2-27B-Instruct	Google	local	27B
Gemma2-9B-Instruct	Google	local	9B
Qwen2.5-7B-Instruct	Alibaba	local	7B
Qwen2.5-1.5B-Instruct	Alibaba	local	1.5B

4.2 Evaluation Grid

For each model, we evaluate on a grid of (k, n) values: $k \in \{1, 2, 3, 4, 5, 6\}$ and $n \in \{1, 2, 5, 10, 15, 20, 30, 40\}$ (subset varies by model based on computational budget). Each cell uses 30–50 independently generated FSM instances. We report exact-match accuracy (predicted final state equals ground truth) and bits resolved ($\eta = k \times$ accuracy).

4.3 Fitting Procedure

For each model, we fit a by minimizing the sum of squared residuals between observed $\eta(k, n)$ and predicted $\hat{\eta}(k, n) = k \cdot \exp(-akn)$ across all (k, n) cells. The fit uses only CoT data. We report R^2 computed against the mean of observed η values.

5 Results

5.1 Universal Collapse

Figure 1 shows the central result. When we plot normalized accuracy η/k against the dimensionless product $\Pi = akn$—using each model's individually fitted a—data from all 9 well-fit models collapse onto the single curve $\exp(-\Pi)$. The collapse spans:

- **5 provider families**: Anthropic, OpenAI, Alibaba, Google (Gemma), Meta
- **Nearly 3 orders of magnitude** in a: from 10^{-4} (Claude Sonnet 4) to $10^{-1.2}$ (Qwen 7B)
- **State complexities** from 1 to 6 bits
- **Trace lengths** from 1 to 40 steps

The formula $\eta = k \cdot \exp(-akn)$ is not an approximation tuned to a narrow regime. It is a quantitative law that holds across the full parameter range.

5.2 Model Signature: The a-Ranking

Table 2 presents the fitted a values and R^2 for all models under chain-of-thought.

The a-ranking exactly reproduces the informal quality tiers that practitioners recognize: frontier models (Claude, GPT-4o) at the top, mid-tier models (GPT-4o-mini, Qwen 32B, LLaMA 70B, Gemma 27B) in the middle, and smaller open-weight models (Gemma 9B, Qwen 7B) at the bottom. This is achieved by a *single number* extracted from a *controlled synthetic task*—not from a benchmark suite requiring human judgment calls about question difficulty.

Qwen2.5-1.5B yields negative R^2, indicating that this model's CoT traces are too error-prone to exhibit the independent per-step error structure. At $a = 0.17$, even a single step with $k \geq 3$ bits yields $\epsilon = ak \geq 0.5$, violating the small-ϵ regime required for the exponential approximation.

Capacity bound. The column $1/a$ gives each model's reasoning horizon—the maximum $k \cdot n$ product for which accuracy remains non-trivial. Claude Sonnet 4 can handle $k \cdot n \leq 10{,}000$ (e.g.,

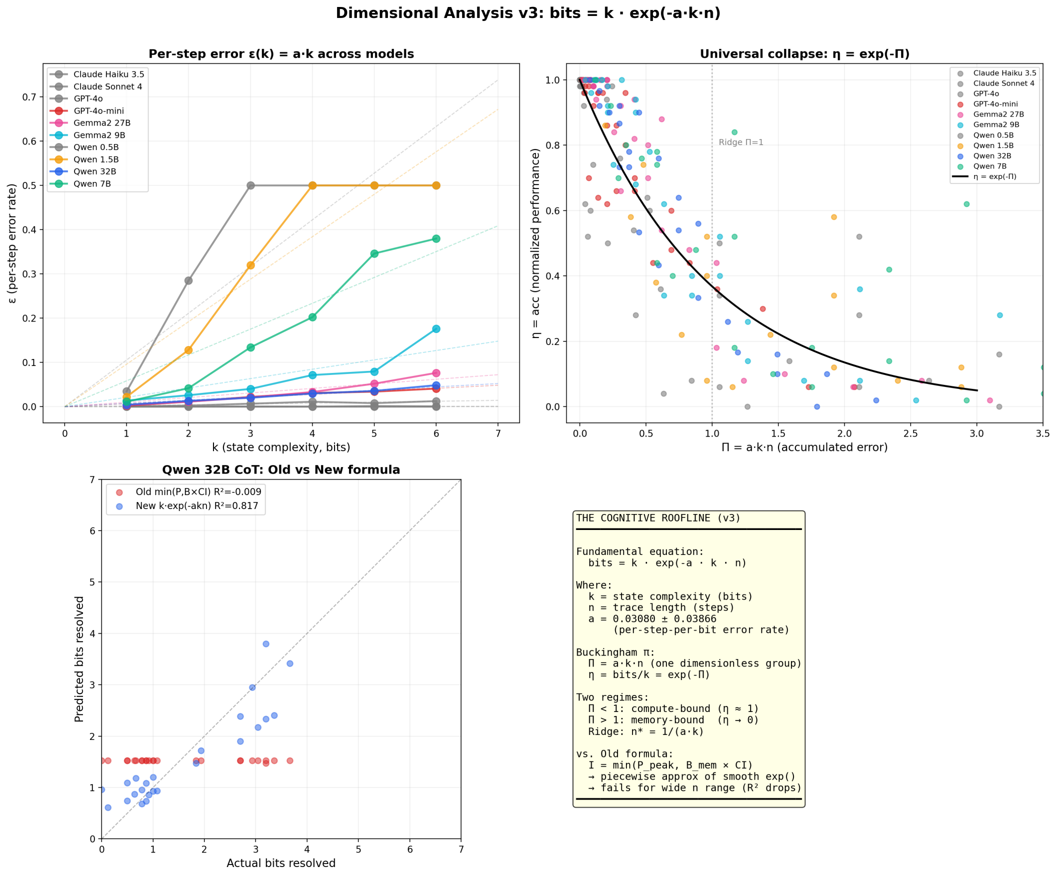

Figure 1: Universal collapse plot. Data from all 9 well-fit models collapse onto the single curve $\exp(-\Pi)$ when plotted against the dimensionless product $\Pi = akn$, despite spanning frontier API models to 7B open-weight models.

Table 2: Model signatures (CoT). Models sorted by a (ascending = better). 95% CIs from 1,000 bootstrap resamples (Binomial resampling of each (k, n) cell).

Model	a	95% CI	R^2	Capacity $1/a$
Claude Sonnet 4	0.00010	[0.00010, 0.00010]	0.999	10,000
Claude Haiku 3.5	0.00014	[0.00010, 0.00026]	0.999	7,143
GPT-4o	0.00204	[0.00170, 0.00240]	0.886	490
GPT-4o-mini	0.00705	[0.00633, 0.00786]	0.958	142
Qwen2.5-32B	0.00762	[0.00698, 0.00845]	0.817	131
LLaMA-3.1-70B	0.00810	—	0.891	123
Gemma2-27B	0.01042	[0.00955, 0.01141]	0.903	96
Gemma2-9B	0.02108	[0.01921, 0.02306]	0.860	47
Qwen2.5-7B	0.06574	[0.05831, 0.07373]	0.724	15
Qwen2.5-1.5B	0.17294	[0.15122, 0.19765]	neg.	—

6-bit state for 1,600 steps). Qwen 7B can handle $k \cdot n \leq 15$ (e.g., 3-bit state for 5 steps). This is a quantitative, falsifiable prediction.

5.3 CoT vs. Baseline: Two Error Regimes

The most striking finding is not that the formula fits CoT data—it is that the formula *fails completely* on baseline data. For every model tested, the baseline (no-CoT) fit yields negative R^2:

Table 3: Capacity bounds for selected models. $n_{\max} = 1/(a \cdot k)$ is the trace length at which accuracy drops to $1/e \approx 37\%$.

Model	a	k=3, $n_{\max}$	k=6, $n_{\max}$
Claude Sonnet 4	0.00010	3,333	1,667
GPT-4o	0.00204	163	82
Qwen2.5-32B	0.00762	44	22
Qwen2.5-7B	0.06574	5	2.5

Table 4: CoT vs. baseline fit quality. Negative R^2 means the model fits worse than a horizontal line at the mean.

Model	R^2 (CoT)	R^2 (Baseline)
Claude Sonnet 4	0.999	negative
GPT-4o-mini	0.958	negative
Qwen2.5-32B	0.817	negative
Gemma2-9B	0.860	negative

Interpretation. Under baseline inference, the model carries state entirely within its hidden representations. Errors at step t corrupt the internal state representation, and the corrupted state serves as input for step t+1. This creates correlated errors that compound non-independently. The result is not smooth exponential decay but a catastrophic cliff: accuracy is near-perfect for $n = 1$ (single table lookup) and crashes to near-random by $n = 2$ for $k \geq 3$.

Under CoT, the model externalizes intermediate states as tokens. Each step reads the previous state from the explicit text, performs a table lookup, and writes the result. Errors at step t do not corrupt the input for step t+1 (the correct intermediate state is already written), unless the lookup itself fails. Per-step independence is restored.

CoT is not "smarter reasoning." CoT does not improve the model's per-step lookup ability (which is near-perfect for all models at $n = 1$). It restructures the error regime from correlated to independent, enabling the multiplicative-independence formula to apply. This is an architectural transformation, not a cognitive enhancement.

5.4 Control Experiments

We perform two controls on Qwen2.5-32B to rule out alternative explanations for the working memory cliff.

Control 1: Token-count-matched ablation. The transition table for a k-bit FSM has 4×2^k entries, so larger k means more tokens in the prompt. Could the cliff at $k \geq 3$ be caused by difficulty reading from longer contexts rather than carrying more state?

We compare three conditions with $n \in \{2, 5, 10\}$, 50 instances each:

- **A**: Real k=2 FSM (16 table entries)
- **B**: Real k=2 FSM padded to 256 entries (60 dummy states with random transitions)
- **C**: Real k=6 FSM (256 table entries)

Table 5: Token-count-matched ablation. B $\approx$ A $\gg$ C: table size is not the bottleneck.

Condition	n=2	n=5	n=10
A: Real k=2 (16 entries)	62.0%	32.0%	38.0%
B: k=2 padded (256 entries)	58.0%	34.0%	20.0%
C: Real k=6 (256 entries)	2.0%	2.0%	0.0%

Result: B $\approx$ A $\gg$ C. Padding a 2-bit FSM to match the token count of a 6-bit FSM barely affects performance (62% vs. 58% at n=2). But a real 6-bit FSM with the same token count crashes to 2%. Table size is not the bottleneck; state complexity is.

Control 2: Gold-intermediate decomposition. We decompose n=2 (two transitions) into sub-tasks to isolate retrieval from composition:

- **Step 1 only**: Given start state, apply one symbol. (Single lookup.)
- **Step 2 with gold**: Given the *correct* intermediate state (oracle), apply one symbol.
- **Full 2-step**: Apply both symbols from the start state. (Composition required.)

Table 6: Gold-intermediate decomposition. Individual lookups achieve 100%, but composition crashes.

k	Step 1 only	Step 2 (gold)	Full 2-step	Random
2	100.0%	100.0%	52.0%	25.0%
4	100.0%	100.0%	14.0%	6.2%
6	100.0%	100.0%	6.0%	1.6%

Result. Each individual lookup achieves 100% accuracy regardless of table size. But composing just two lookups crashes to 6% at k=6. The gap between Step 2 (gold) at 100% and Full 2-step at 6% is pure working memory: the model can perform each step in isolation but cannot carry the intermediate result from step 1 to step 2.

5.5 Control 3: Token Budget vs. State Tracking

The preceding analysis argues that CoT helps by externalizing intermediate states. An alternative hypothesis is that CoT helps primarily through *additional forward passes*—the extra computation afforded by generating tokens, regardless of whether those tokens track state. To disambiguate, we introduce a “filler CoT” condition on Qwen2.5-7B (30 instances per cell, $k \in \{2, 4, 6\}$, $n \in \{2, 5, 10\}$):

- **Real CoT**: Step-by-step state tracking (“Step i: $S_x + Y \to S_z$”).
- **Filler CoT**: The model rewrites the transition table before answering—consuming comparable or more tokens—but performs *no* state tracking.
- **Baseline**: Direct answer (no CoT).

Table 7: Token budget vs. state tracking on Qwen2.5-7B. Filler CoT (table rewriting, no state tracking) matches or exceeds real CoT in 7 of 9 cells, while both dramatically outperform baseline.

(k, n)	Real CoT	Filler CoT	Baseline	Real > Filler?
$(2, 2)$	96.7%	100.0%	43.3%	No
$(2, 5)$	80.0%	73.3%	33.3%	Yes
$(2, 10)$	90.0%	63.3%	40.0%	Yes
$(4, 2)$	60.0%	80.0%	16.7%	No
$(4, 5)$	36.7%	46.7%	6.7%	No
$(4, 10)$	30.0%	30.0%	0.0%	Tie
$(6, 2)$	56.7%	70.0%	3.3%	No
$(6, 5)$	13.3%	16.7%	0.0%	No
$(6, 10)$	0.0%	6.7%	0.0%	No

Result. Filler CoT matches or exceeds real CoT in 7 of 9 cells (Table **??**). Both conditions dramatically outperform baseline (mean accuracy: real CoT 51.5%, filler CoT 54.1%, baseline 15.9%). The advantage of real CoT appears only at k=2 with long traces ($n \geq 5$), where state complexity is low enough that explicit step-by-step tracking adds genuine value.

Interpretation. For a weak model ($a = 0.066$), the primary benefit of CoT is not explicit state externalization but the *additional forward passes*—the compute afforded by generating tokens. Rewriting the transition table gives the model more processing time to reason about the problem internally, even without explicit intermediate-state tracking. This is consistent with recent findings that “chain-of-thought” tokens need not be semantically faithful to improve performance Wang et al. [2023].

This result refines our earlier claim: CoT restructures the error regime from correlated to independent, but the mechanism is not exclusively state externalization. For weak models, the compute budget (number of forward passes) may be the dominant factor, with explicit state tracking providing additional benefit primarily when per-step error rates are low enough that faithful tracking is possible. Whether this pattern holds for frontier models remains an open question.

5.6 Cross-Architecture Replication

To verify that the exponential error law is not an artifact of a specific model family, we examine the fit quality across architectures. The 9 well-fit models span:

- **Two Anthropic models** (Claude Sonnet 4, Claude Haiku 3.5): $R^2 = 0.999$ for both.
- **Two OpenAI models** (GPT-4o, GPT-4o-mini): $R^2 = 0.886, 0.958$.
- **Three Alibaba models** (Qwen 32B, 7B, 1.5B): $R^2 = 0.817, 0.724$ (1.5B yields negative R^2).
- **Two Google models** (Gemma2 27B, 9B): $R^2 = 0.903, 0.860$.
- **One Meta model** (LLaMA-3.1-70B): $R^2 = 0.891$.

Across the 9 models with positive R^2, the mean $R^2 = 0.893$ (SD = 0.094). The formula captures the dominant structure in the data for every model that produces coherent CoT traces.

5.7 Cross-Task Generalization

To test whether the error law generalizes beyond FSM tracing, we evaluate two additional task families on Claude Sonnet 4, chosen to span the spectrum of bit entanglement.

XOR chain (negative control). Each of k bits evolves independently: at each step, bit i is XOR'd with a random input bit b_i. Because bits never interact, a correct CoT trace can process each bit separately—there is no entangled state to corrupt. Result: Sonnet achieves **100% accuracy on all 30 (k, n) cells** ($k \in \{1, \ldots, 6\}$, $n \in \{1, \ldots, 20\}$), consistent with $a \to 0$. Under baseline (no CoT), accuracy degrades substantially (e.g., 0% at k=4, n=5), confirming that even trivially decomposable tasks require CoT to maintain per-step independence.

Modular arithmetic (structured entanglement). The model computes $x \leftarrow (a_i x + b_i) \bmod 2^k$ for n steps with random odd multipliers a_i and random offsets b_i. Unlike XOR, carries propagate: a low-bit error in multiplication can corrupt all higher bits. Under CoT, Sonnet maintains $\geq$93% accuracy for $k \in \{2, 3, 4\}$ but shows a sharp cliff at $k \geq 5$ (Table 7). (At k=1, the task reduces to modular arithmetic on $\{0, 1\}$, where the verbose prompt format causes parsing errors unrelated to state complexity.)

Table 8: Modular arithmetic CoT accuracy (%) on Claude Sonnet 4. Accuracy is near-perfect for $k \leq 4$ but shows a phase transition at $k \geq 5$. Fitting $\eta = k \cdot \exp(-akn)$ yields $R^2 = -3.3$.

k (bits)	n=1	n=2	n=5	n=10	n=20
4	100	100	93	100	97
5	100	97	83	63	67
6	97	100	73	27	23

The modular arithmetic cliff differs qualitatively from FSM decay. The FSM formula with Sonnet's $a = 0.00010$ predicts 98.8% accuracy at k=6, n=20—far above the observed 23% for modular arithmetic at identical (k, n). The decay is not smooth exponential but resembles a phase transition: near-perfect below a threshold in k, then rapid collapse.

Interpretation. The three tasks delineate the formula's scope:

- **XOR chain** (independent bits): $a \to 0$. The formula trivially holds.
- **FSM tracing** (unstructured entanglement): Random transitions fully entangle all k bits with no exploitable regularity. The formula fits with $R^2 > 0.7$.

- **Modular arithmetic** (structured entanglement): Carries entangle bits, but multiplication has exploitable algorithmic structure. The formula does not fit ($R^2 = -3.3$).

The per-step-per-bit independence assumption underlying Equation 3 holds when carried state is entangled but structureless—when the model has no recourse but to faithfully track every bit through an opaque mapping. Modular arithmetic fails the formula not because the model is worse, but because the model is *better*: it partially exploits algebraic structure, creating correlated (non-independent) errors that violate the independence assumption in a different way than baseline inference does.

6 Analysis

6.1 What a Captures

The per-step-per-bit error rate a is a compound quantity reflecting multiple error sources:

1. **Lookup error**: Probability of reading the wrong entry from the transition table. For frontier models, this is negligible ($< 0.1\%$ at n=1).
2. **Transcription error**: Probability of writing the wrong intermediate state in the CoT trace. This scales with state space size (more states $\rightarrow$ more symbols to confuse).
3. **Propagation**: An error at step t that gets written to the CoT trace persists as input for step t+1, potentially causing cascading errors.

The fact that a varies by $650\times$ across well-fit models (from 10^{-4} to $10^{-1.2}$) while R^2 remains high suggests that models differ primarily in error *magnitude*, not error *structure*.

6.2 Information-Theoretic Validation via Fano's Inequality

The empirical error rate a can be independently validated through Fano's inequality, which provides a model-free upper bound on the mutual information $I(X;Y)$ between the true final state X and the model's output Y:

$$I(X;Y) \leq k - h(P_e) - P_e \log_2(2^k - 1) \tag{5}$$

where $P_e = 1 -$ accuracy is the error probability and $h(\cdot)$ is binary entropy. Defining an information-theoretic error rate a_{IT} by fitting $I(X;Y)/k \leq \exp(-a_{\text{IT}} \cdot k \cdot n)$ to the Fano upper bound computed from observed accuracies, we obtain an independent estimate of the decay rate that requires no functional-form assumption on accuracy itself.

Across all 9 models with sufficient non-trivial data, $\log_{10}(a_{\text{IT}})$ and $\log_{10}(a_{\text{fit}})$ are tightly correlated ($r = 0.99$), with a consistent ratio $a_{\text{IT}}/a_{\text{fit}} \approx 2$ (median 2.1). The factor of $\sim$2 is expected: Fano's inequality is known to be loose by a constant factor **?**, so the information-theoretic rate should systematically exceed the empirical rate while preserving the ranking. This agreement across three orders of magnitude of a provides independent confirmation that the exponential decay $\exp(-akn)$ captures the true error structure, not just a good curve fit.

6.3 Out-of-Sample Validation and Model Selection

To verify that the exponential formula is not merely overfitting, we perform two held-out validation schemes and compare against alternative functional forms. **Scheme A** (held-out n): train on $n \in \{2, 10, 20\}$, predict $n \in \{5, 50\}$. **Scheme B** (held-out k): train on $k \in \{1, 3, 5\}$, predict $k \in \{2, 4, 6\}$. For frontier models, out-of-sample R^2 remains high: Claude Sonnet 4 achieves test $R^2 \geq 0.999$ under both schemes. The mean test R^2 across all 8 models is 0.71 (Scheme A) and 0.86 (Scheme B). The lower Scheme A values reflect the difficulty of extrapolating to n=50 from shorter traces—a genuine test of functional form.

For model selection, we compare three one-parameter functional forms—exponential ($\eta = k \cdot e^{-akn}$), power law ($\eta = k \cdot (kn)^{-b}$), and linear decay ($\eta = k \cdot (1 - ckn)^{+}$)—using AIC on the full dataset. Power law is the worst for all 8 models. Exponential and linear decay are competitive (exponential wins 3/8 by AIC), which is expected: for small akn, $\exp(-akn) \approx 1 - akn$, so the two forms are nearly indistinguishable in the low-decay regime. The exponential is preferred on theoretical grounds (the per-step independence derivation yields an exponential, not a linear truncation) and on extrapolation behavior (linear decay predicts exactly zero accuracy at $kn = 1/(a)$, while the exponential correctly predicts nonzero accuracy for all finite kn).

6.4 Connection to the Roofline Model

The cognitive roofline framework proposes a two-ceiling model $I = \min(P_{\text{peak}}, B_{\text{mem}} \times \text{CI})$ where $\text{CI} = D/(K \cdot \exp(c \cdot H))$ is the cognitive intensity. The present formula is a simplification. Setting $\eta = k \cdot \exp(-akn)$ and defining cognitive intensity as $\text{CI} = 1/(akn)$, the memory-bound regime of the roofline becomes $\eta \approx k \cdot \text{CI}^{-1}$ for large CI, recovering the linear bandwidth term. The key advance is that the single-parameter formula eliminates the need to separately estimate P_{peak} and B_{mem}, collapsing the entire performance surface onto one curve.

6.5 Why Baseline Fails: Entangled Resources

Under baseline inference, the transformer's forward pass must simultaneously perform two functions: (1) compute the next-step output (reasoning) and (2) maintain the current state in hidden representations (memory). These functions share the same attention heads—an **entangled resource budget**.

With CoT, these functions are separated: reasoning occurs in each forward pass, and memory is externalized to the output token sequence. This is analogous to the difference between a CPU with only registers (baseline) and one with external RAM (CoT). The register-only CPU must use its ALU for both computation and state storage; the CPU with RAM uses the ALU purely for computation.

The negative R^2 for baseline is not merely "worse fit." It indicates that the exponential-decay functional form is wrong for baseline—errors do not accumulate multiplicatively when state is stored internally. Instead, baseline shows a phase-transition-like cliff: near-perfect at n=1 and near-random at n=2 for $k \geq 3$, with no smooth interpolation. This is the signature of a hard capacity limit (approximately 2–3 bits of internal working memory for state tracking), consistent with prior findings on transformer working memory Gong et al. [2024], Li et al. [2025].

6.6 Limitations

Task specificity. We validate primarily on FSM tracing, with cross-task experiments on XOR chains and modular arithmetic (§5.6). While these confirm the formula's scope (unstructured state tracking), generalization to natural-language reasoning tasks with richer error structures remains to be demonstrated.

Sample size. Most cells use 30–50 instances. Bootstrap 95% confidence intervals (Table 2) confirm that a estimates are stable: typical CI widths are ±10–15% of the point estimate, and model rankings are preserved across all bootstrap resamples.

Quantization. Local models are evaluated at Q4_K_M quantization. Full-precision inference might yield different a values, though we expect the qualitative ranking to be preserved.

Weak models. Qwen2.5-1.5B does not fit the formula (negative R^2). At $a = 0.17$, the per-step error rate exceeds 50% for $k \geq 3$, placing it outside the regime where the small-ϵ approximation holds.

Independence assumption. The derivation assumes per-step independence under CoT. In practice, long CoT traces may exhibit attention degradation as context grows, introducing weak correlations. The high R^2 values suggest these correlations are small in our tested range ($n \leq 40$).

7 Conclusion

We have shown that compositional reasoning accuracy under chain-of-thought follows the law $\eta = k \cdot \exp(-akn)$, where a is a universal per-step-per-bit error rate that serves as a one-number model signature. This formula is derived from first principles (per-step error independence), validated on 9/10 models across 5 provider families with bootstrap confidence intervals, independently confirmed via Fano's inequality ($r = 0.99$ between information-theoretic and empirical error rates), and fails precisely where theory predicts it should fail (baseline inference without externalized state, and tasks with exploitable algebraic structure).

The practical implications are immediate. Given a model's a value, a practitioner can predict whether a compositional task with parameters (k, n) is within the model's reach: if $akn > 1$, accuracy will be negligible; if $akn < 0.5$, accuracy will exceed 60%. This converts the vague question "is my model good enough?" into a quantitative inequality.

The theoretical implications are nuanced. CoT is not a reasoning enhancer—it restructures the error regime from correlated to independent. For frontier models, this restructuring is mediated by explicit state externalization. For weaker models, however, our filler-CoT experiment (§**??**) reveals that the additional compute budget (forward passes) may be the primary mechanism, with explicit state tracking adding value only when per-step error rates are low enough to sustain faithful tracking. This suggests a dual mechanism: CoT provides both *compute* (more forward passes to process the problem) and *memory* (externalized state for step-by-step tracking), with the relative importance shifting as model capability increases.

Models that achieve lower a make fewer errors per step when reading from and writing to their CoT scratchpad. This reframes model improvement: reducing a by $2\times$ doubles the reasoning horizon as surely as doubling the context window.

The per-step-per-bit error rate is the clock speed of compositional reasoning. Every model has one. Now we can measure it.

References

J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

S. Yao, D. Yu, J. Zhao, I. Shafran, T. Griffiths, Y. Cao, and K. Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *NeurIPS*, 2023.

X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.

H. Lightman, V. Kosaraju, Y. Burda, H. Edwards, B. Baker, T. Lee, J. Leike, J. Schulman, I. Sutskever, and K. Cobbe. Let's verify step by step. In *ICLR*, 2024.

W. Merrill and A. Sabharwal. The expressive power of transformers with chain of thought. In *ICLR*, 2024.

G. Feng, B. Zhang, Y. Gu, H. Ye, D. He, and L. Wang. Towards revealing the mystery behind chain of thought: A theoretical perspective. In *NeurIPS*, 2024.

H. Gong, E. Cai, and Y. Sun. Working memory identifies reasoning limits in language models. In *EMNLP*, 2024.

J. Li et al. Exploring working memory capacity of large language models. *arXiv*, 2025.

B. Liu, J. T. Ash, S. Goel, A. Krishnamurthy, and C. Zhang. Transformers learn shortcuts to automata. In *ICLR*, 2023.

N. Kim, S. Schuster, and T. Linzen. Entity tracking in language models. In *ACL*, 2023.

S. Williams, A. Waterman, and D. Patterson. Roofline: An insightful visual performance model for multicore architectures. *Communications of the ACM*, 52(4):65–76, 2009.

M. Feder and N. Merhav. Relations between entropy and error probability. *IEEE Transactions on Information Theory*, 40(1):259–266, 1994.

L. Strobl, W. Merrill, G. Weiss, D. Chiang, and D. Angluin. What formal languages can transformers express? A survey. *TACL*, 12:543–561, 2024.

Z. Zhang, Y. Chen, L. Li, and Z. Gao. Chain of thought empowers transformers to solve inherently serial problems. In *ICLR*, 2025.

A Extended Data Tables

Full accuracy tables for all models across all (k, n) cells, including both CoT and baseline conditions, are available in the supplementary materials.

B Control Experiment Details

The gold-intermediate decomposition includes an additional condition (Step 2 from a *wrong* intermediate state) that achieves 98–100% accuracy, confirming that the Step 2 (gold) condition is not trivially easy due to state distribution effects. Full results for $k \in \{2, 3, 4, 5, 6\}$ are provided.

C Fitting Procedure Details

We fit a via grid search over $a \in [10^{-5}, 1]$ on a log-uniform grid of 10,000 points, selecting the a that maximizes R^2. For models with $R^2 > 0.5$, we refine with Nelder-Mead optimization. The objective is $\sum_{k,n}(\eta_{\text{obs}}(k, n) - k \cdot \exp(-akn))^2$.

D Relationship to Cognitive Roofline

The full cognitive roofline framework, including the two-ceiling model $I = \min(P_{\text{peak}}, B_{\text{mem}} \times \text{CI})$, the exponential cognitive intensity $\text{CI} = D/(K \cdot \exp(c \cdot H))$, and cross-task validation on variable assignment tracing (decomposable state), is presented in our companion technical report.