COT-DEPTH: INSTANCE-LEVEL SERIAL DEPTH GOVERNS CHAIN-OF-THOUGHT BUDGET

Anonymous authors
Paper under double-blind review

ABSTRACT

Chain-of-thought (CoT) theory assigns a single worst-case budget to every instance of a problem class, yet sequential difficulty within a class spans orders of magnitude: on s-t connectivity, a star graph resolves in one step while a length-$(n-1)$ path demands $\Theta(n)$ steps at the same encoding length. We introduce *instance serial depth* $d(x)$—the minimum depth of a computation-trace DAG under a canonical rule system—as the first per-instance CoT complexity measure, and prove matching upper and lower bounds on the CoT budget in terms of $d(x)$. A constant-depth log-precision Transformer solves any instance x in $d(x)+O(\log n)$ CoT steps, extending to $\lceil d(x)/L \rceil + O(\log n)$ for L layers. Within the bounded-state, local-prefill class $\mathcal{M}^{\mathrm{loc}}_{L,b}$, correctness on pointer-chasing instances forces $\Omega(d(x)/L)$ steps unconditionally across the full depth range, and full prompt-to-prompt attention preserves the linear rate. Class-level budgeting therefore wastes $\Omega(n/\log n)$ steps relative to instance-adaptive allocation—a **depth–size separation**. Experiments on three synthetic task families (s-t connectivity, multi-digit addition, Boolean formula evaluation) confirm d-linear scaling in every one of 21 model×task configurations ($p < 0.02$), with the per-instance slope reaching $+1.07$ in a $4\times$ wider model. A depth probe on hidden states cuts MAE by up to 67% over length baselines and, plugged into an instance-adaptive allocator, saves 19.6% of inference steps on STCON at 100% accuracy. The law extends beyond training from scratch: a LoRA-fine-tuned Qwen2.5-7B carries d-linear step counts to templated multi-hop QA, and off-the-shelf 7–8B instruction models allocate depth-proportional CoT under mere few-shot prompting—Qwen3-8B emits $+0.79$ CoT steps per unit depth (R^2=0.54, $p < 10^{-100}$), independently reproduced on Qwen2.5-7B ($+0.43$, $p < 10^{-43}$).

1 INTRODUCTION

Chain-of-thought (CoT) prompting lets language models solve problems beyond the reach of their fixed-depth architecture by generating intermediate reasoning tokens (Wei et al., 2022; Kojima et al., 2022). Recent theory formalizes this power at the *class* level: T CoT steps give constant-depth Transformers the power of circuits with $T(n)$ sequential layers, lifting them from TC^0 to uniform P (Li et al., 2024; Merrill & Sabharwal, 2024). Yet every instance in the class inherits the same $T(n)$ budget, regardless of its actual sequential difficulty.

Sequential difficulty varies by orders of magnitude within a single problem class. On s-t connectivity (STCON), a star graph in which t neighbors s resolves in one step, while a length-$(n-1)$ path demands $\Theta(n)$ steps—same problem, same encoding length (Figure 1). A uniform budget sized for the path wastes $n-2$ steps on the star; one sized for the star fails on the path. No existing theory predicts per-instance CoT requirements.

Instance serial depth. We isolate this per-instance quantity as *instance serial depth* $d(x)$: given a canonical rule system $\mathcal{R}$ for the task family—breadth-first reachability for STCON, ripple carry for addition, short-circuit evaluation for Boolean formulas—$d(x)$ is the longest chain of necessarily sequential steps, equivalently the minimum depth of a complete computation-trace DAG for x. Unlike input length, $d(x)$ varies within a family: the same Boolean formula under different variable assignments spans depths from $O(1)$ (top-level short circuit) to $\Theta(n)$ (full evaluation).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

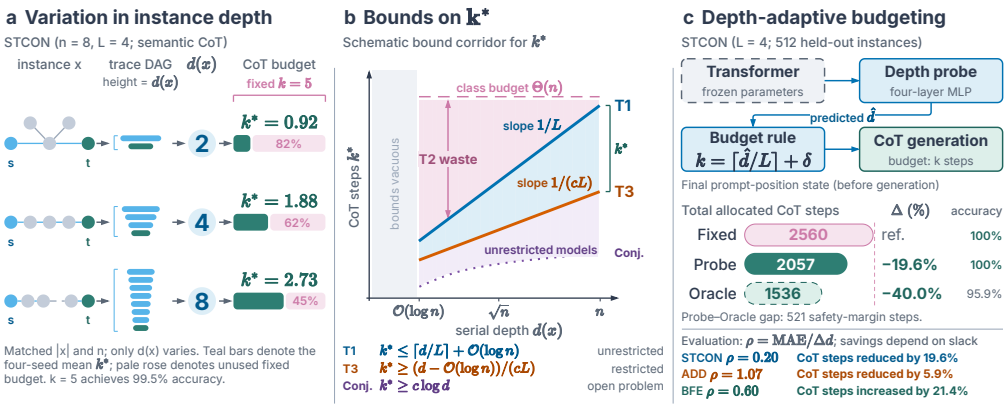


Figure 1: **Instance serial depth $d(x)$ governs per-instance CoT budget.** **(a)** Three STCON instances at depths $d(x) \in \{2, 4, 8\}$: class-level budgeting wastes 45–82% of steps on easier instances (rose bars). **(b)** The T1 ceiling $(1/L)$ and T3 floor $(1/cL)$ sandwich the achievable budget; the gap to the class-level line is the $\Omega(n/\log n)$ waste of Theorem 3.15. **(c)** A depth probe predicts $\hat{d}(x)$ and sets per-instance budget; deployability turns on $\rho = \mathrm{MAE}/\Delta d$. On STCON ($\rho$=0.20) this yields 19.6% savings at 100% accuracy; on ADD/BFE ($\rho \geq 0.60$) the safety margin absorbs gains.

Matching upper and lower bounds. We pin the CoT budget to $d(x)$ from both sides. A constant-depth Transformer solves any x in $d(x) + O(\log n)$ steps (Theorem 3.6); an L-layer model in $\lceil d(x)/L \rceil + O(\log n)$ (Corollary 3.7). Conversely, within the bounded-state, local-prefill class $\mathcal{M}^{\mathrm{loc}}_{L,b}$, correctness on pointer-chasing instances forces $\Omega(d(x)/L)$ steps (Theorem 3.10); the bound holds unconditionally across the full depth range $d(x) \leq \Theta(n)$—in particular in the deep regime that earlier round-elimination arguments could not reach. Full prompt-to-prompt attention preserves the linear rate; only the constant degrades (Proposition 3.14). Two knobs delimit this regime exactly: bounded state (C1) and composition-hard tasks (C3); relaxing the width knob motivates a conjectured $\Omega(\log d_{\mathcal{R}}(x))$ envelope for unrestricted poly-width models. Together, the two bounds imply that class-level budgeting wastes $\Omega(n/\log n)$ steps relative to instance-adaptive allocation, with the gap growing in input length (Theorem 3.15).

Empirical evidence. Experiments on the three synthetic task families confirm d-linear scaling in every one of 21 model×task configurations, with one-sided t-tests rejecting a zero slope at $p < 0.02$ throughout. A depth probe on hidden states predicts $d(x)$ well enough to save 19.6% of inference steps on STCON at 100% accuracy via instance-adaptive allocation. The law transfers up the scale ladder: a fine-tuned 7B LLM carries d-linear step counts to templated multi-hop QA across 4 seeds (Appendix O), and off-the-shelf 7–8B instruction models already allocate depth-proportional CoT under few-shot prompting—Qwen3-8B emits $+0.79$ CoT steps per unit depth (R^2=0.54, $p < 10^{-100}$), independently reproduced on Qwen2.5-7B ($+0.43$, $p < 10^{-43}$; Appendix N.2).

Our contributions.

- **An instance-level complexity measure with a matching per-instance upper bound.** We define instance serial depth $d(x)$ via a canonical rule system (Definition 3.2), prove it is polynomial-time computable (Proposition 3.3), and show it coincides with the shallowest-evidence sub-DAG depth in any bounded-fan-in circuit for the task (Lemma C.1). A constant-depth log-precision Transformer solves any x in $d(x) + O(\log n)$ CoT steps (Theorem 3.6); an L-layer variant in $\lceil d(x)/L \rceil + O(\log n)$ (Corollary 3.7), upgrading the class-level $T(n)$ budget of Li et al. (2024); Merrill & Sabharwal (2024) to a per-instance $T(x)$.
- **The first multi-layer, per-instance linear lower bound, with a depth–size separation.** Within a precisely delimited class $\mathcal{M}^{\mathrm{loc}}_{L,b}$ of bounded-state hard-attention Transformers, pointer-chasing instances force $\Omega(d(x)/L)$ steps unconditionally in n, k, H, L, b (Theorem 3.10); C1–C3 tightly characterize the boundary, and full prompt-to-prompt prefill preserves linearity (Proposition 3.14). The upper and lower bounds sandwich the optimum and imply an $\Omega(n/\log n)$ waste of class-level budgeting on $\mathcal{X}^{\mathrm{ptr}}_n$ (Theorem 3.15).

- **Empirical validation across scale, and a practical probe.** Across three task families and 21 configurations, trained Transformers exhibit d-linear scaling ($p<0.02$); a hidden-state depth probe recovers $d(x)$ well enough to save 19.6% of inference steps on STCON at 100% accuracy. The law transfers up the scale ladder: fine-tuned GPT-2 (Appendix N.1) and a fine-tuned 7B LLM on templated multi-hop QA (Appendix O) both preserve the linear rate, and off-the-shelf Qwen3-8B and Qwen2.5-7B allocate depth-proportional CoT under mere few-shot prompting (Appendix N.2).

2 Related Work

Class-level CoT expressiveness. Prior theory bounds CoT length as a function of *input length*. Feng et al. (2023) give the first constructive simulation showing constant-depth Transformers with $T(n)$ CoT tokens capture $T(n)$-time serial computation; Li et al. (2024); Merrill & Sabharwal (2024) sharpen this into tight simulations between T CoT steps and T-layer bounded-precision circuits, characterizing the $\mathrm{TC}^0 \to \mathrm{P}$ ladder as $T(n)$ grows. Every bound is stated at the class level—one $T(n)$ per input length. Instance serial depth $d(x)$ collapses to $\Theta(n)$ on the worst-case slice these results describe but is $O(1)$ on the majority of instances in the same family—exactly the gap Theorems 3.6 and 3.10 pin down.

Per-architecture and pointer-chasing lower bounds. Existing CoT lower bounds share a single instance grain—worst case per family—and are typically single-layer or single-architecture. Peng et al. (2024) prove $\Omega(\sqrt{n})$ steps for a single-layer Transformer on iterated composition; Sanford et al. (2024) draw a correspondence between CoT and Massively Parallel Computation; Amiri et al. (2025) prove UHAT-class $\Omega(|E|\log|V|)$ bounds for reachability. Theorem 3.10 moves the frontier along the axis that matters for adaptive computation: to our knowledge, it is the first per-instance, multi-layer linear CoT lower bound, holding across the full depth range $d_{\mathcal{R}}(x) \le \Theta(n)$ within a precisely delimited L-layer class $\mathcal{M}^{\mathrm{loc}}_{L,b}$ (App. D). The two grains are complementary: worst-case class-level bounds and per-instance bounds attack orthogonal questions.

Instance-adaptive length allocation. A parallel empirical line adapts inference budget per instance without formal grounding: TALE (Han et al., 2025) and SmartThinker (He et al., 2025) train difficulty predictors on rollouts and truncate CoT to a learned length; inference-compute scaling treats budget as a global knob (Snell et al., 2025; Brown et al., 2024); filler and pause tokens (Goyal et al., 2024; Pfau et al., 2024) explore semantically empty budget. Each of these supplies the training *regime* that induces adaptive stopping and treats difficulty as an opaque scalar regressed from outputs; CoT-Depth supplies the missing *measure*—$d_{\mathcal{R}}(x)$ is computed in polynomial time from x (Proposition 3.3), is model-independent, and is the target Theorems 3.6–3.10 pin the optimal budget to on both sides, transferring across scales and pretraining regimes (Appendices N.1, O, N.2). Pairing this instance-grained measure with matching per-instance upper (T1) and multi-layer lower (T3) bounds and a class-vs-instance separation (T2) turns adaptive length allocation from a heuristic into a theoretically anchored primitive. Broader coverage—MPC and UHAT variants, overthinking, reasoning-by-superposition—lives in App. A.

3 Theoretical Framework: Instance Serial Depth and CoT Complexity

Setting. We consider non-uniform families of decoder-only Transformers of constant depth $L_0 = O(1)$ (extended to L layers in Corollary 3.7), width $m = \mathrm{poly}(n)$, log-precision arithmetic, and causal attention, following Merrill & Sabharwal (2024); Li et al. (2024). Given input $\mathrm{enc}(x)$, the model emits a thinking prefix of $T = T(x)$ CoT tokens followed by an answer. Prior theory bounds the *class-level* budget $T(n) := \max_{x\in\mathcal{X}_n} T(x)$; we characterize the per-instance budget $T(x)$. A *rule system* $\mathcal{R}$ (Definition 3.1 below) is the semantic backbone against which per-instance difficulty is measured: it names the atomic reasoning moves for a task family—adding two digits with carry, propagating a reachability flag along an edge, evaluating a conjunction—as a finite catalogue of bounded-arity inference rules over a fact universe $\mathcal{F}_n$ of polynomial size. For each family we fix a *canonical* rule system (breadth-first reachability for STCON, ripple carry for ADD, short-circuit

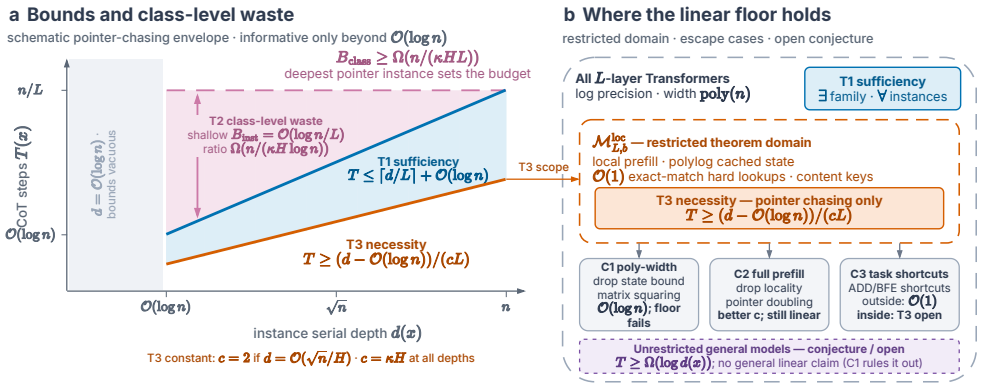


Figure 2: **Where the serial-depth bounds bite, and where they do not.** $d \equiv d_{\mathcal{R}}(x)$; $T(x)$ = CoT steps; $c = \kappa H$ (sharpening to 2 for $d = O(\sqrt{n})$). **(a)** Schematic envelope (axes qualitative). Blue ceiling: sufficiency $T(x) \leq \lceil d/L \rceil + O(\log n)$ (Theorem 3.6). Orange floor: necessity $T(x) \geq (d - O(\log n))/(cL)$ inside $\mathcal{M}^{\text{loc}}_{L,b}$ (Theorem 3.10), holding across the full range. Grey band: $d = O(\log n)$ where the sandwich is vacuous. Rose line: class-level waste $\Omega(n/\log n)$ (Theorem 3.15). **(b)** Scope: the floor is a statement about $\mathcal{M}^{\text{loc}}_{L,b}$. Poly-width collapses cost to $O(\log n)$; algebraic shortcuts to $O(1)$; contentful prefill degrades only the constant (Proposition 3.14). For unrestricted Transformers we conjecture $T(x) \geq c \log d$ (Conjecture L.1).

evaluation for BFE; App. G); canonicity is required for the upper bound and for empirical prediction, but Theorem 3.10 is rule-system free (Corollary 3.11).

3.1 Formalization: Rule Systems and Instance Serial Depth

Definition 3.1 (Rule system)**.** *A rule system is a tuple $\mathcal{R} = (\mathcal{F}_n, I, O, \Delta)$ for each input size n, where $\mathcal{F}_n$ is a finite fact universe with $|\mathcal{F}_n| = \text{poly}(n)$; $I : \mathcal{X}_n \to 2^{\mathcal{F}_n}$ maps each input x to its axiom set (computable coordinate-locally from x); $O \subseteq \mathcal{F}_n$ is the set of output facts; and Δ is a finite set of inference rule schemas of bounded arity $O(1)$. An instantiated rule (P, f) has premises $P \subseteq \mathcal{F}_n$ with $|P| \leq k$ and conclusion $f \in \mathcal{F}_n$. Rule validity is* local*: decidable from the premises and $O(1)$ input-fact coordinates named by the schema. $\mathcal{R}$ must be sound (every derivable output fact yields the correct answer) and complete (some output fact is always derivable).*

Definition 3.2 (Computation trace and instance serial depth)**.** *A* computation trace *for x under $\mathcal{R}$ is a DAG $\mathcal{T}$ whose sources are a subset of $I(x)$ and whose every non-source node f has in-neighbors P forming an instantiated rule $(P, f) \in \Delta(x)$. A trace is* complete *if it contains output facts sufficient to determine the full answer $f(x)$. The* instance serial depth *of x is*

$$d_{\mathcal{R}}(x) \;:=\; \min\big\{\, \text{depth}(\mathcal{T}) : \mathcal{T} \textit{ is a complete trace for } x \textit{ under } \mathcal{R} \,\big\}, \tag{1}$$

where $\text{depth}(\mathcal{T})$ counts the maximum number of non-source nodes on any directed path. When the rule system $\mathcal{R}$ is clear from context, we write $d(x)$ for $d_{\mathcal{R}}(x)$.

Proposition 3.3 (Poly-time computability)**.** *$d_{\mathcal{R}}(x)$ is computable in $O(|\Delta(x)| \cdot k + |\mathcal{F}_n| \log |\mathcal{F}_n|)$ time via priority-queue relaxation over the derivation hypergraph.*

Proof sketch. The monotone recursion $\text{depth}^*(f) = 1 + \min_{(P,f)\in\Delta(x)} \max_{p\in P} \text{depth}^*(p)$ admits Dijkstra-style closure.

$d_{\mathcal{R}}(x)$ equals the depth of the shallowest *evidence sub-DAG* (equivalently, the shallowest complete trace) in any bounded-fan-in circuit computing f (Lemma C.1, Appendix C), which connects the measure directly to the circuit frameworks of Li et al. (2024) and Merrill & Sabharwal (2024).

Remark 3.4 (Rule system dependence and canonicity)**.** *For each family we fix the* canonical *rule system—the breadth-first or data-parallel algorithm matching T1's construction. Shortcut systems such as transitive closure ($d_{\text{tc}}(x) = O(\log \text{dist}_G(s,t))$) exist but require $\Theta(n^2)$ state per round, exceeding the $O(nbL)$ budget of $\mathcal{M}^{\text{loc}}_{L,b}$—precisely the poly-width mechanism scoped out by C1. The*

necessity bound itself is rule-system free (Corollary 3.11); canonicity is required only for T1 and for $d_{\mathcal{R}}$ to be an accurate empirical predictor (ΔAIC rejects shortcut models on every measured configuration; Appendix Q).

Remark 3.5 (Canonical depth formulas)**.** *STCON: $d(x) = \mathrm{dist}_G(s,t)+1$ (YES) or $\mathrm{ecc}_G(s) + \lceil \log_2 n \rceil + 3$ (NO); ADD: $d(x) = \mathrm{maxrun}_P(x) + O(1)$; BFE: shallowest evidence sub-DAG depth. Full definitions in Appendix G.*

3.2 Sufficiency: Upper Bound on CoT Steps

We work with non-uniform families of constant-depth, log-precision decoder-only Transformers of $\mathrm{poly}(n)$ width, following Li et al. (2024).

Theorem 3.6 (Sufficiency — T1)**.** *Fix a task family with canonical rule system $\mathcal{R}$ (bounded arity, $|\mathcal{F}_n| = \mathrm{poly}(n)$). There exists a constant-depth Transformer family $\{M_n\}$ (weights depending on n but not on x) such that for every instance $x \in \mathcal{X}_n$, greedy decoding from prompt $\mathrm{enc}(x)$ produces the correct answer with thinking budget*

$$T(x) \;\le\; d_{\mathcal{R}}(x) + c \cdot \log n, \tag{2}$$

where c depends only on $\mathcal{R}$.

Corollary 3.7 (Depth scaling — T1′)**.** *For every $L \ge 1$, there exists an L-layer family[1] with $T(x) \le \lceil d_{\mathcal{R}}(x)/L \rceil + c \cdot \log n$.*

Proof sketch (full proof in Appendix B.1). Let $D_i = \{f \in \mathcal{F}_n : \mathrm{depth}^*(f) \le i\}$ collect the facts derivable within trace depth $\le i$, so $D_0 = I(x)$ and $D_{i+1} = \Phi_x(D_i)$ for a depth-2 monotone circuit Φ_x (OR-of-ANDs over bounded-arity rules). We construct $\{M_n\}$ so that thinking position p_i holds the indicator of D_i: the first position assembles D_0 via one attention layer; each subsequent position attends to p_{i-1} and applies Φ_x via two MLP layers; generation halts when an output fact appears, at step $d_{\mathcal{R}}(x)$, with the additive $c \cdot \log n$ absorbing assembly and halting overhead. For T1′, stack L closure blocks within one position to advance L rounds per token.

From class-level to per-instance budgets. Building on the CoT simulation template of Li et al. (2024), Theorem 3.6 tightens the class-level budget $T(n)$ into a per-instance budget $d_{\mathcal{R}}(x)$, dissolving the order-of-magnitude spread that a class-level bound conceals.

3.3 Necessity: Per-Instance Linear Lower Bound

Two knobs delimit the regime where per-instance linear necessity holds—one architectural (width), one task-side (composition-hardness)—and the linear rate is robust to the most realistic prefill relaxation.

Proposition 3.8 (Boundary of the linear regime)**.**

- ***C1.*** Width is the architectural knob. *At width $\Theta(n^2)$, iterated Boolean matrix squaring completes every STCON instance in $O(\log n)$ steps; bounded-state models are therefore the natural regime for a per-instance linear floor, and a matching $\Omega(\log d_{\mathcal{R}}(x))$ envelope for unrestricted poly-width models is conjectured (Conjecture L.1).*
- ***C2.*** Prefill contents affect only the constant *(Proposition 3.14): moving from local to full prompt-to-prompt prefill—the most realistic relaxation—preserves linearity and buys at most an exponential-in-L constant.*
- ***C3.*** Composition is the task-side knob. *Composition-hard pointer chasing forces $\Omega(d_{\mathcal{R}}(x)/L)$ steps; the algebraic families ADD (carry-lookahead) and BFE (short-circuit) admit $O(1)$-step solutions independent of $d_{\mathcal{R}}(x)$, and family-specific bounds for them are a natural extension (Appendix F).*

C1 and C3 identify the two knobs (width, composition-hardness) that jointly delimit the regime; C2 shows the most realistic prefill relaxation lies inside it. The class $\mathcal{M}^{\mathrm{loc}}_{L,b}$ we define next captures this regime precisely.

[1] We use "depth" in four distinct senses throughout: (i) *instance serial depth* $d(x)$ (sequential complexity of an input), (ii) *model depth* L (number of Transformer layers), (iii) *DAG depth* in the computation trace, and (iv) *extraction layer* for probe experiments. Context disambiguates.

Definition 3.9 (Restricted class $\mathcal{M}^{\mathrm{loc}}_{L,b}$)**.** *A model $M \in \mathcal{M}^{\mathrm{loc}}_{L,b}$ satisfies: (1)* local prefill*—the KV entry of prompt position j is a fixed function of (x_j, j) only (no prompt-to-prompt attention); (2)* bounded generation*—when emitting token i, the model runs L layers with $H = O(1)$ exact-match content-addressed hard-attention lookups per layer, each addressing $\kappa = O(1)$ positions and returning b bits; cached state per position is $\le b$ bits per layer.*

Theorem 3.10 (Necessity — T3)**.** *Let $\mathcal{X}^{\mathrm{ptr}}_n$ be the pointer-chasing instance family (directed graphs on $[n]$ with planted chain of length k, balanced YES/NO endpoint; full specification in Appendix D). Fix $b = \mathrm{polylog}(n)$ and let every $M \in \mathcal{M}^{\mathrm{loc}}_{L,b}$ use* exact-match *content-only attention keys, with $H = O(1)$ heads each addressing $\kappa = O(1)$ cells per lookup. If M is correct on $\mathcal{X}^{\mathrm{ptr}}_n$, then for every $x \in \mathcal{X}^{\mathrm{ptr}}_n$ with chain length $k \le (n-3)/2$:*

- ***(i)*** Unconditional linear floor, every depth. *With* no *relation assumed among n, k, H, L, b (cell counting, Theorem D.5),*

$$T_M(x) \;\ge\; \frac{d_{\mathcal{R}}(x) - \mathcal{O}(\log n)}{c \cdot L}, \qquad c = \kappa H = O(1), \tag{3}$$

  so the linear floor holds across the full range $d_{\mathcal{R}}(x) \le \Theta(n)$—in particular in the deep regime $k = \Theta(n)$ that the earlier round-elimination argument could not reach.
- ***(ii)*** Sharpened constant in the sparse-id regime. *For $k = O(\sqrt{n}/H)$ the constant improves to $c = 2$ (round elimination, Theorem D.7).*

Proof sketch (full proofs in Appendix D). A correct M induces an r-round ($r = LT$) adaptive exact-match probe protocol with $\le bL$ bits of state between tokens and H queries per round (Lemma D.1). *Tier (i).* Each probe returns ≤ 1 input cell, so r rounds see $\le Hr$ cells. On the hard distribution the $k-1$ "aligned" chain/decoy cell-pairs are pairwise disjoint, and a suffix-swap involution rerouting any *unseen* pair flips the YES/NO label while leaving the transcript unchanged; if $Hr \le k-2$ some pair is always unseen, every transcript fibre is label-balanced, and success is exactly $1/2$. Hence $LT \ge (k-1)/(\kappa H)$, with no further hypothesis (Theorem D.5). *Tier (ii).* For $k = O(\sqrt{n}/H)$ lucky content-key hits are rare, and inductive round elimination over the Nisan–Wigderson distribution (Nisan & Wigderson, 1993) removes one pointer per two rounds, giving $LT \ge (k - O(1))/2$ (Theorem D.7).

Corollary 3.11 (The floor is rule-system free)**.** *Theorem 3.10(i) is a statement about the instance and the model, not about $\mathcal{R}$: the cell-counting proof lower-bounds $L \cdot T_M(x)$ by $(k-1)/(\kappa H)$, where k is the planted chain length of $x \in \mathcal{X}^{\mathrm{ptr}}_n$, and never evaluates a rule system—$d_{\mathcal{R}}(x)$ enters* (3) *only through the identity $d_{\mathcal{R}}(x) = k + \mathcal{O}(\log n)$ for the canonical BFS system. Hence for* every *sound rule system $\mathcal{R}'$, including shortcut systems with $d_{\mathcal{R}'}(x) = \mathcal{O}(\log d_{\mathcal{R}}(x))$ such as transitive closure, the floor $T_M(x) \ge (k-1)/(\kappa H L)$ holds verbatim. A shortcut rule system cannot be used to evade the bound; it can only fail to* predict *it, since it reports $\mathcal{O}(\log k)$ where the true requirement is $\Omega(k/L)$. Canonicity is needed for the* upper *bound (T1's construction simulates $\mathcal{R}$ round by round) and for $d_{\mathcal{R}}$ to be a usable predictor—not for necessity.*

Remark 3.12 (Open directions)**.** *The $c = \kappa H$ floor is unconditional for all $k \le (n-3)/2$; the sharper $c = 2$ is proven for $k = O(\sqrt{n}/H)$ only—extending it is provably impossible for $H \ge 3$ in the unbounded-state abstraction (Appendix D).*

Remark 3.13 (Content-key restriction)**.** *The exact-match content-key condition is essential: a positional sweep reads the cell array in $\lceil (2k-1)/H \rceil$ rounds, beating the bound for $H \ge 5$ (Appendix R).*

Proposition 3.14 (Contentful-prefill robustness)**.** *Let $\widetilde{\mathcal{M}}_{L,b}$ extend $\mathcal{M}^{\mathrm{loc}}_{L,b}$ by replacing local prefill with full prompt-to-prompt attention during prefill (causal prefill is a subcase), retaining exact-match content-only keys, $H = O(1)$ lookups per layer addressing $\kappa = O(1)$ positions each, and $b = \mathrm{polylog}(n)$ cached bits per position per layer.*

- ***(i)*** Linearity survives. *Every $M \in \widetilde{\mathcal{M}}_{L,b}$ correct on $\mathcal{X}^{\mathrm{ptr}}_n$ obeys, for every x with chain length $k \le (n-3)/2$,*

$$T_M(x) \;\ge\; \frac{d_{\mathcal{R}}(x) - \mathcal{O}(\log n) - (1+\kappa H)^L}{c_p \cdot L}, \qquad c_p = \kappa H (1+\kappa H)^L, \tag{4}$$

  unconditionally in n, k, H, L, b (the subtracted $(1+\kappa H)^L$ is $O(1)$ at fixed architecture).

(ii) Doubling matches it. *There exists* $M \in \widetilde{\mathcal{M}}_{L,b}$ *with* $T_M(x) \leq \lceil d_{\mathcal{R}}(x)/(2^L{-}1)\rceil + \lceil (2^L{-}1)/L\rceil + O(1)$.

Proof sketch (full proof in Appendix E). (i) A layer-ℓ generation state depends on at most $(1{+}\kappa H)^\ell$ transcript cells (induction: each lookup adds κH cells, each dragging its layer-$(\ell{-}1)$ dependency set), so T tokens expose at most $(TLH\kappa{+}1)(1{+}\kappa H)^L$ cells; the suffix-swap involution of Theorem D.5 then applies with this larger visibility budget. (ii) During prefill, position p's layer-ℓ state stores a jump of span 2^ℓ; each generation token composes $2^0{+}\cdots{+}2^{L-1} = 2^L{-}1$ hops, and a one-hop-per-layer window scan covers the remainder.

Summary: matching bounds sandwich the optimum. Combining Corollary 3.7 with Theorem 3.10(i) pins the optimal step count on $\mathcal{X}_n^{\mathrm{ptr}}$ between two linear functions of $d_{\mathcal{R}}(x)$:

$$\frac{d_{\mathcal{R}}(x) - \mathcal{O}(\log n)}{c \cdot L} \;\leq\; T^*_{\mathcal{M}}(x) \;\leq\; \left\lceil \frac{d_{\mathcal{R}}(x)}{L} \right\rceil + \mathcal{O}(\log n), \tag{5}$$

informative whenever $d_{\mathcal{R}}(x) = \omega(\log n)$ (Figure 2a). Proposition 3.14 shows contentful prefill—the most realistic relaxation of $\mathcal{M}^{\mathrm{loc}}_{L,b}$—moves the constant from $\Theta(\kappa H)$ to $\Theta(\kappa H(1{+}\kappa H)^L)$ but preserves the linear order. Beyond bounded state and hard attention, the framework extends to a matching $\Omega(\log d_{\mathcal{R}}(x))$ bound for unrestricted poly-width Transformers (Conjecture L.1, Appendix L).

3.4 Depth–Size Separation

For each task family, subfamilies at identical encoding length $\Theta(n)$ span $d(x)$ from $O(1)$ to $\Theta(n)$; for BFE, the same formula at different variable assignments traverses all three regimes (Appendix F).

Theorem 3.15 (Depth–size separation — T2)**.** *On* $\mathcal{X}_n^{\mathrm{ptr}}$ *within* $\mathcal{M}^{\mathrm{loc}}_{L,b}$*, the multiplicative waste of class-level over instance-adaptive allocation is* $\Omega(n/(\kappa H \log n))$ *for constant* L*—unconditional across the full depth range.*

T1 holds for all three families; T3 applies to STCON-type composition-hard instances, matched to C3's characterization of where linear necessity holds. Rule-system-specific bounds for ADD (carry-lookahead) and BFE (short-circuit) are natural extensions (Table 1, Appendix F).

The characterized regime. T1 is a constructive existence result for constant-depth log-precision Transformers of $\mathrm{poly}(n)$ width; T3 pins the necessity side within $\mathcal{M}^{\mathrm{loc}}_{L,b}$ and its contentful-prefill extension $\widetilde{\mathcal{M}}_{L,b}$, forcing $\Omega(d_{\mathcal{R}}(x)/L)$ steps on pointer-chasing instances unconditionally in n, k, H, L, b. C1–C3 map the frontier: bounded state is the architectural regime where linear necessity provably holds, full prompt-to-prompt prefill lies inside it (Prop. 3.14), and composition-hard tasks are the task-side regime. Above the width knob, the same information-theoretic core (resolving a k-hop chain requires reading $\Omega(\log k)$ serialized positions) motivates a matching $T_M(x) \geq c \log d_{\mathcal{R}}(x)$ envelope for unrestricted poly-width Transformers (Conjecture L.1, Appendix L). Empirical validation of the linear rate follows in §4.

4 Experiments

Because $d(x)$ is exactly computable (Proposition 3.3), the theory yields directly testable predictions without any learned difficulty proxy: on task families where $d(x)$ controls the sequential requirement, the per-instance CoT budget should scale linearly with $d(x)$, the slope should track per-step throughput, and $d(x)$ should be predictable well enough from hidden states to yield real inference savings. We test each in turn. (i) The stable minimal budget $k^*(x)$ scales linearly with $d(x)$ across families and model depths (§4.2), and the dependence survives a supervision-side control that decouples trace length from $d(x)$ (§4.4). (ii) The linear coefficient tracks per-step throughput: it grows with L on STCON and reaches slope $+1.07$ under a $4\times$ wider model (§4.3). (iii) A hidden-state probe recovers $d(x)$ well enough to convert the linear law into concrete inference savings (§4.5).

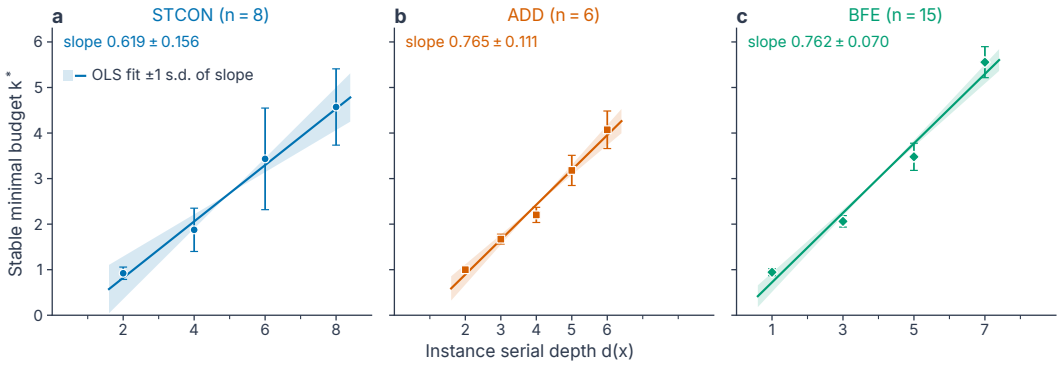


Figure 3: Stable minimal budget k^* vs. $d(x)$ (L=4, semantic mode; points are depth strata; error bars ± 1 s.d. across 4 seeds; full coverage in Appendix J). The linear trend indicates that $d(x)$ controls per-instance CoT requirements in each family.

4.1 Experimental Setup

We use three synthetic families where $d(x)$ is exactly computable: **STCON** (n=8 vertices), directed s-t connectivity; **ADD** (n=6 digits), base-10 addition; **BFE** (n=15 gates), Boolean formula evaluation on caterpillar formulas. Each family produces balanced training sets with $d(x)$ spanning $[1, \Theta(n)]$, and we verify all generators are free of exploitable shortcuts beyond the canonical algorithm's sequential structure (Appendix H). We train decoder-only Transformers ($L \in \{1, 2, 4, 8, 12\}$, $\sim$3.2M parameters at L=4, 4 heads, d_{model}=256) with teacher-forced CoT traces to convergence over 12k steps; a 3$\times$ extended run confirms convergence. Two CoT modes bracket the information spectrum: *semantic* traces externalize meaningful intermediate facts, whereas *filler* traces consist entirely of content-free padding tokens before the answer.

The key metric is the per-instance *stable minimal budget* $k^*(x)$: the smallest budget k such that the model answers correctly for all $k' \geq k$,[2] reported as the mean over instances with $d(x) = d$ via a full sweep over the budget.[3] All 21 grid configurations (plus n-scaling runs at n=12/16/31/32/64-planted) use 4 seeds; error bars are mean $\pm$ std across seeds. Full coverage and controls (prompt-length confounds, teacher-forcing artifacts) appear in Appendices J and T.

4.2 Minimal Steps Scale Linearly with $d(x)$

Figure 3 displays the core relationship: across all three families at L=4, measured k^* rises steadily with $d(x)$, with fitted slopes of 0.619 ± 0.156 (STCON, YES-verdict instances), 0.765 ± 0.111 (ADD), and 0.762 ± 0.070 (BFE). Every one of the 21 model$\times$task configurations yields a strictly positive slope, and one-sided t-tests reject a zero slope at $p < 0.02$ throughout (Tables 6 and 7, Appendices J and J.1).

We compute STCON slopes on YES-verdict instances because NO instances carry a $+\lceil \log_2 n \rceil + 3$ bookkeeping offset in $d_{\mathcal{R}}$ that creates Simpson's paradox when pooled (Remark 3.5). d-linear scaling holds at larger n: at n=16 the slope is 0.552 ± 0.090 and at n=32 it is 0.231 ± 0.050, both strictly positive (Appendix J).

4.3 The Slope Tracks Per-Step Throughput

All 21 configurations across $L \in \{1, 2, 4, 8, 12\}$ yield strictly positive slopes (Figure 4, Appendix J), and the pattern across the grid confirms the theory's curriculum–capacity factorization. The three families share a single training signal (one canonical rule step per emitted token), so extra depth pays exactly when the per-step primitive is a bottleneck: STCON's primitive composes an unbounded

[2]Formally, $k^*(x) = \min\{k : \text{model correct at all } k' \geq k\}$; results are robust to first-success and 95%/99.5% threshold variants (Appendix J.2).

[3]When the per-stratum violation rate is below 2%, a subset uses binary search (first-success k^* only); first-success and stable slopes agree to within 5% for all configurations (Appendix J.2).

neighborhood lookup with a set-union write, and its slope rises with L ($0.36 \rightarrow 0.79$); ADD's and BFE's bounded-arity boolean updates already express one rule step at L=1, so Corollary 3.7's $\lceil d_{\mathcal{R}}/L \rceil$ envelope predicts a slope insensitive to L—and that is exactly what the trained models show. Width scaling—enlarging the per-step vocabulary rather than the step count—sharpens the effect on the harder family: the $4\times$ wider 13M STCON model reaches slope $+1.074$ (YES-only) at n=8 and remains strictly positive under a uniform budget cap at n=64 ($+0.200 \pm 0.088$; Figure 6). The law is architecture-independent: fine-tuning GPT-2 (124M) yields YES-slope 0.74 ± 0.09 ($p < 0.001$; Appendix N.1). Across every trained model, empirical step counts sit in the achievability envelope of Corollary 3.7 and above the necessity floor of Theorem 3.10 (Appendix I).

4.4 The Linear Law Rides on the Semantic Trace, Not on Length

To isolate the semantic channel from any trace-length effect, we replace meaningful intermediate tokens with content-free filler (Goyal et al., 2024; Pfau et al., 2024) and hold filler count constant, uniform-random, or mean-matched to the depth-proportional arm. Severing the length–$d(x)$ correlation collapses the slope to ≤ 0.026 ($\geq 98\%$ reduction across all three families) while accuracy stays at 98.5–100% (Appendix T): the linear law is carried by depth-proportional semantic content, not by trace length. Two additional findings on the same models: the *depth absorption* effect (minimal steps decrease with L, saturating beyond L=4; Appendix T) and the stride result ($k^* \propto 1/s$), confirming that the counted unit is forward passes rather than tokens.

4.5 From Law to Inference Savings

A 2-layer MLP probe on frozen hidden states at the final prompt position recovers $d(x)$ without access to the reasoning trace: probe gain (fractional MAE reduction over a prompt-length baseline) reaches 67% on STCON and 50% on BFE, and filler-trained models encode $d(x)$ at least as strongly as semantic ones (Figure 7, Appendix J). Feeding these predictions into an instance-adaptive allocator (with a 99.5%-accuracy safety margin) on L=4 semantic models converts the linear law of §4.2 into concrete inference savings: 100% accuracy while saving 19.6% of steps on STCON and 5.9% on ADD (Table 9, Figure 9, Appendix K). Beyond the savings themselves, the deployability ratio $\rho = \mathrm{MAE}/\Delta d$ is a *falsifiable a-priori prediction* of when adaptive allocation pays: computed from probe MAE alone, it correctly ranks all three families ($\rho = 0.20$ / 0.60 / 1.07 predicting large savings / marginal / none), instantiating the class-vs-instance separation of Theorem 3.15 as a deployment criterion.

5 Conclusion

Reasoning difficulty is an instance-level property, and instance serial depth $d(x)$ makes the cost of ignoring it concrete: $d(x)$ sets how many steps a model needs, explains why instances of identical length differ dramatically in difficulty, and quantifies how much a class-level budget wastes. The theory is tight up to architecture constants—the optimal budget on pointer-chasing instances is sandwiched between $(d(x) - O(\log n))/(\kappa HL)$ and $\lceil d(x)/L \rceil + O(\log n)$, linearity survives full prompt-to-prompt attention (Proposition 3.14), and an $\Omega(n/\log n)$ separation quantifies class-level waste. Experiments confirm d-linear scaling in every one of 21 configurations ($p < 0.02$), and a hidden-state depth probe converts the law into 19.6% inference savings on STCON at 100% accuracy—the payoff Theorem 3.15 predicts and Table 9 realizes. Taken together, these results refine the class-level $T(n)$ characterization (Li et al., 2024; Merrill & Sabharwal, 2024) into a per-instance $T(x)$ that trained Transformers already track and a small hidden-state probe can recover.

The law transfers up the scale ladder. d-linear scaling survives every jump we test: a fine-tuned 124M GPT-2 reproduces the slope (App. N.1), a fine-tuned 7B model carries it to templated multi-hop QA across 4 seeds (App. O), and off-the-shelf Qwen3-8B ($+0.79$, R^2=0.54) and Qwen2.5-7B ($+0.43$) allocate depth-proportional CoT under few-shot prompting (App. N.2). The same framework opens two directly programmed extensions: an $\Omega(\log d_{\mathcal{R}}(x))$ envelope for unrestricted poly-width models (Conjecture L.1) and family-specific bounds for ADD (carry-lookahead) and BFE (short-circuit) with an explicit stopping criterion for the overthinking regimes documented by Chen et al. (2024); Sui et al. (2025).

USE OF AI ASSISTANTS

Generative AI assistants (large language models) were used in three capacities while preparing this work. First, for prose polishing and LaTeX formatting assistance. Second, for retrieval and discovery, including surveying and locating related work. Third, AI assistants contributed to deriving and proving the paper's mathematical claims—the instance-serial-depth complexity measure, the matching upper and lower bounds on the chain-of-thought budget in terms of it, the resulting depth–size separation, and the supporting proofs in the appendix. The authors independently verified every derivation and proof step, and confirmed the resulting predictions against the trained-model experiments (21 model$\times$task configurations) and the frontier-model measurements reported in the paper; the experimental design and analysis of results are the authors' own. The authors take full responsibility for the final content of this work, including any material produced with the aid of generative AI.

ETHICS STATEMENT

All experiments use fully synthetic data and involve no human subjects, private information, or deployed systems.

REPRODUCIBILITY STATEMENT

The supplement provides complete source code, data generators, and hyperparameter specifications (Appendix H); all trained-model experiments use fixed seeds with 4-seed verification; frontier-model evaluations report instance-bootstrap 95% confidence intervals (Appendix N).

REFERENCES

Alireza Amiri, Xinting Huang, Mark Rofin, and Michael Hahn. Lower bounds for chain-of-thought reasoning in hard-attention transformers. In *International Conference on Machine Learning*, 2025.

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

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

Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. Towards revealing the mystery behind chain of thought: A theoretical perspective. In *Advances in Neural Information Processing Systems*, 2023.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Sharan Narang. Think before you speak: Training language models with pause tokens. In *International Conference on Learning Representations*, 2024.

Michael Hahn. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, 2020.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

Xingyang He, Xiao Ling, and Jie Liu. Smartthinker: Learning to compress and preserve reasoning by step-level length control. *arXiv preprint arXiv:2507.04348*, 2025.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, pp. 22199–22213, 2022.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. In *International Conference on Learning Representations*, 2024.

William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. In *International Conference on Learning Representations*, 2024.

Noam Nisan and Avi Wigderson. Rounds in communication complexity revisited. *SIAM Journal on Computing*, 22(1):211–219, 1993.

Binghui Peng, Srini Narayanan, and Christos Papadimitriou. On limitations of the transformer architecture. *arXiv preprint arXiv:2402.08164*, 2024.

Jacob Pfau, William Merrill, and Samuel R. Bowman. Let's think dot by dot: Hidden computation in transformer language models. *arXiv preprint arXiv:2404.15758*, 2024.

Clayton Sanford, Daniel Hsu, and Matus Telgarsky. Transformers, parallel computation, and logarithmic depth. In *International Conference on Machine Learning*, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. In *International Conference on Learning Representations*, 2025.

Philip M. Spira. On time-hardware complexity tradeoffs for boolean functions. In *Proceedings of the Fourth Hawaii International Symposium on System Sciences*, pp. 525–527, 1971.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.

Hanlin Zhu, Shibo Hao, Zhiting Hu, Jiantao Jiao, Stuart Russell, and Yuandong Tian. Reasoning by superposition: A theoretical perspective on chain of continuous thought. In *Advances in Neural Information Processing Systems*, 2025.

A Extended Related Work

We expand on the related-work discussion in §1, covering additional connections beyond the core positioning already addressed there.

Massively Parallel Computation (MPC). Sanford et al. (2024) draw a formal connection between CoT-augmented Transformers and the MPC model, showing that T-step CoT corresponds to T rounds of local computation with bounded communication. Our framework is compatible: instance serial depth $d(x)$ can be read as the minimum MPC round complexity of instance x under the bounded-arity rule system, connecting per-instance CoT requirements to per-instance MPC depth.

Iterated composition and pointer chasing. Peng et al. (2024) prove $\Omega(\sqrt{n})$ CoT steps for 1-layer models on iterated function composition—a worst-case bound on a specific architecture. Our pointer-chasing lower bound (Theorem 3.10) extends this in two directions—per-instance granularity and multi-layer applicability—at the cost of restricting to $\mathcal{M}^{\mathrm{loc}}_{L,b}$.

Unique Hard-Attention Transformers (UHAT). Amiri et al. (2025) prove CoT lower bounds for UHAT models, including $\Omega(|E| \log |V|)$ for graph reachability. Their model class allows contentful prefill and positional addressing—relaxations our $\mathcal{M}^{\mathrm{loc}}_{L,b}$ excludes—yielding stronger worst-case bounds but without per-instance granularity. Neither result subsumes the other: ours is per-instance and multi-layer; theirs is worst-case under a broader model class.

Overthinking and early stopping. Chen et al. (2024) and Sui et al. (2025) find that additional CoT steps can *hurt* performance. Our framework assumes monotonic benefit (more budget $\geq$ less error), which holds in the synthetic setting. Extending $d(x)$-based allocation to overthinking regimes would require coupling the depth measure with a stopping criterion.

Reasoning by superposition. Zhu et al. (2025) propose maintaining continuous latent states rather than discrete token chains, achieving sub-linear step counts through width—analogous to our C1 counterexample (poly-width matrix squaring in $O(\log n)$ steps).

B Full Proofs

B.1 Proof of Theorem 3.6 (Sufficiency)

We restate the theorem and give the complete proof.

Theorem (T1, restated)**.** *Fix a task family with canonical rule system $\mathcal{R} = (\mathcal{F}_n, I, O, \Delta)$ of bounded arity $k = O(1)$ and $|\mathcal{F}_n| = \mathrm{poly}(n)$. There exists a non-uniform family of constant-depth ($L_0 = O(1)$ layers), log-precision, $\mathrm{poly}(n)$-width decoder-only Transformers $\{M_n\}$ such that for every n and every instance $x \in \mathcal{X}_n$, greedy decoding produces the correct answer with thinking budget $T(x) \leq d_{\mathcal{R}}(x) + c \cdot \log n$.*

B.2 Model Specification

Definition B.1 (Constant-depth log-precision decoder)**.** *A decoder-only Transformer M with:*

- *$L_0 = O(1)$ layers, width $m = O(|\mathcal{F}_n|) = \mathrm{poly}(n)$, fixed head count;*
- *Log-precision arithmetic: all activations are rationals in $O(\log n)$ bits (Merrill & Sabharwal convention);*
- *Causal attention with KV-cache: position p at layer ℓ reads layer-$(\ell-1)$ outputs of positions $\leq p$;*
- *Greedy decoding; vocabulary: input symbols, $\langle \mathrm{step} \rangle$, answer symbols, $\langle \mathrm{eos} \rangle$.*

B.3 Closure Operator

Let $D_i := \{f \in \mathcal{F}_n : \mathrm{depth}^*(f) \leq i\}$. Define the one-round closure:

$$\Phi_x(S) = S \cup \{f : \exists (P, f) \in \Delta(x) \text{ with } P \subseteq S\}.$$

Then $D_0 = I(x)$ and $D_{i+1} = \Phi_x(D_i)$.

Lemma B.2 (Uniformity of Φ_x)**.** *A single monotone Boolean circuit Φ of depth 2 and size $\mathrm{poly}(n)$, independent of x, computes $\mathbf{1}_{\Phi_x(S)}$ from inputs $(\mathbf{1}_S, \mathbf{1}_{I(x)})$.*

Proof. For each fact f: $f \in \Phi_x(S) \Leftrightarrow \bigvee_{(P,f)\in\Delta} \bigwedge_{p\in P}[p \in S \cup I(x)]$. This OR-of-ANDs has depth 2 and total size $O(|\mathcal{F}_n| \cdot |\Delta|) = \mathrm{poly}(n)$. Under log precision with $\{0, 1\}$ inputs, AND/OR are linear-threshold functions computable by a 2-layer MLP.

Lemma B.3 (Input assembly)**.** *An $O(1)$-layer sub-network places $\mathbf{1}_{I(x)} \in \{0, 1\}^{|\mathcal{F}_n|}$ at a designated layer ℓ^* of the first thinking position p_0.*

Proof. One attention layer with uniform attention aggregates one-hot fact encodings from prompt tokens into a sum vector (exact: sums $\leq \mathrm{poly}(n)$, fits in $O(\log n)$ bits per coordinate). One MLP layer thresholds at ≥ 1 to produce the indicator. Each prompt token contributes its coordinate-locally decodable input facts.

Lemma B.4 (Closure step)**.** *If p_{i-1} holds $\mathbf{1}_{D_{i-1}}$ at layer ℓ^*, then p_i can hold $\mathbf{1}_{D_i}$ at ℓ^* using $O(1)$ layers.*

Proof. At p_i: (1) one attention layer with hard attention to p_{i-1} (positional flag) copies $\mathbf{1}_{D_{i-1}}$; same layer copies $\mathbf{1}_{I(x)}$ from p_0. (2) Two MLP layers apply the depth-2 circuit of Lemma B.2, yielding $\mathbf{1}_{D_i}$. The emitted token is $\langle \mathrm{step} \rangle$; information flows through the KV cache.

Lemma B.5 (Halting and readout)**.** *$O(1)$ layers detect success and emit the answer.*

Proof. **Decision tasks:** $D_i \cap O \neq \emptyset$ is an OR gate. Output: $\mathrm{ans}(o)$ for the derived o. **ADD:** All $n{+}1$ facts $\mathrm{Out}(i)$ present is a threshold-$(n{+}1)$ gate. Answer digits read via mod-$(n{+}1)$ position counter over $n{+}1$ answer positions.

B.4 Main Proof

Proof of Theorem 3.6. By Lemma B.3, p_0 holds $\mathbf{1}_{D_0}$. By Lemma B.4 and induction, p_i holds $\mathbf{1}_{D_i}$ for all $i \geq 0$. $D_{d_{\mathcal{R}}(x)}$ contains an output fact (by definition). By Lemma B.5, the model halts at step $d_{\mathcal{R}}(x) + O(1)$ and emits $f(x)$. The $c \cdot \log n$ absorbs constant-factor overhead.

Proof of Corollary 3.7. Stack L copies of Lemma B.4 within one position (residual stream connects blocks). One token advances L rounds: p_i holds D_{iL}. Total tokens: $\lceil d_{\mathcal{R}}(x)/L \rceil + O(1)$.

C Circuit Correspondence

Lemma C.1. *Fix $\mathcal{R}$ with arity k. For every x:*

1. (Trace $\rightarrow$ circuit) *A Boolean circuit C of size $O(|\mathcal{F}_n| \cdot |\Delta|)$ and fan-in k computes f on all inputs; its sub-circuit activated on x has depth $O(d_{\mathcal{R}}(x))$.*
2. (Circuit $\rightarrow$ trace) *Any bounded fan-in circuit C computing f gives $d_{\mathcal{R}_C}(x) =$ depth of the shallowest evidence sub-DAG for $C(x)$.*

Proof. (1) Facts $\rightarrow$ wires; rule instantiations $\rightarrow$ constant-depth gadgets. Layer the minimal trace by depth. (2) Gate evaluations are rules of arity = fan-in. OR with a true child / AND with a false child short-circuits (needs one child). The shallowest evidence DAG = minimal complete trace.

D Proof of Theorem 3.10 (Necessity)

D.1 Protocol Extraction

Lemma D.1. *$M \in \mathcal{M}^{\mathrm{loc}}_{L,b}$ correct on $\mathcal{X}^{\mathrm{ptr}}_n$ with budget T induces a deterministic (LT)-round adaptive query protocol with state $\le bL$ bits between tokens, H queries/round each returning b bits, total information $\le LT \cdot Hb$ bits from the input.*

Proof. Local prefill $\Rightarrow$ prompt KV entries are raw input (no composition). Each generated token = L layers $\times$ H lookups = LH queries against uncomposed input. Cached state between tokens: residual stream $\le bL$ bits. Each hard-attention lookup addresses $O(1)$ positions and returns $\le b$ bits; thus each round (one layer of one token) reveals at most Hb bits about the input, and r rounds reveal at most $r \cdot Hb \le LT \cdot Hb$ bits total.

D.2 Hard Distribution

The pointer-chasing distribution μ on $\mathcal{X}^{\mathrm{ptr}}_n$:

1. Sample a random functional graph on $[n]$ with in-degree ≤ 2 on the s-component.
2. Plant chain $s = v_0 \to v_1 \to \cdots \to v_k$ (random subset of $[n]$, random order).
3. Balanced coin: w.p. $1/2$ set $v_k = t$ (YES); w.p. $1/2$ place t in separate component (NO).

D.3 Cell-Count Bound: the Unconditional Linear Floor

The following bound holds for *every* chain length $k \le (n-3)/2$, with no relation assumed among n, k, H, L, b; it is the source of Theorem 3.10(i). Abstract the prompt as an array of *cells*, one per edge token (u, v). An *exact-match* probe $\mathrm{out}(u)$ (resp. $\mathrm{in}(v)$) returns the unique cell whose source is u (resp. target v), or $\varnothing$ if none. Write μ_k for the pointer distribution restricted to chain length k: the planted s-chain contributes cells $c_i = (u_{i-1}, u_i)$, $i = 1..k$ (with $u_0 = s$, u_k the YES/NO terminal), and the decoy contributes $e_j = (w_j, w_{j+1})$, $j = 1..k{-}1$. Call $\{c_i, e_{i-1}\}$, $i = 2..k$, the $k{-}1$ pairwise-disjoint *aligned pairs*.

Lemma D.2 ($\le$ one cell per probe)**.** *Every vertex of μ_k has in- and out-degree ≤ 1, so each exact-match probe returns ≤ 1 cell; after r rounds the set K_r of returned cells has $|K_r| \le Hr$. (On the in-degree-≤ 2 generator promise, $|K_r| \le 2Hr$ and every constant below doubles.)*

Lemma D.3 (Aligned-pair suffix swap)**.** *For $i \in \{2, \ldots, k\}$ let σ_i exchange the suffixes of the chain and decoy at chain position i and decoy position $i{-}1$. Then: (i) σ_i is a μ_k-measure-preserving involution on $\mathrm{supp}(\mu_k)$ that flips the YES/NO label; (ii) x and $\sigma_i(x)$ agree on every cell except c_i, e_{i-1}, which are rerouted to (u_{i-1}, w_i) and (w_{i-1}, u_i); (iii) the source multiset and the target multiset are unchanged, so a probe* misses *in x iff it misses in $\sigma_i(x)$, and the only keys whose* answer *changes are $\mathrm{out}(u_{i-1}), \mathrm{out}(w_{i-1}), \mathrm{in}(u_i), \mathrm{in}(w_i)$, each returning c_i or e_{i-1}.*

Proof. Cells c_j $(j < i)$ and e_j $(j < i{-}1)$ are untouched; the chain suffix c_j $(j > i)$ reappears as decoy cells and the decoy suffix e_j $(j \ge i)$ as chain cells, so the cell multiset changes only in the two rerouted entries, and the new chain has $i + (k - i + 1) = k + 1$ vertices (length k) ending at the opposite terminal—giving (i)–(ii). For (iii), the source set $\{s, u_1, \ldots, u_{k-1}\} \cup \{w_1, \ldots, w_{k-1}\}$ and target set $\{u_1, \ldots, u_{k-1}, \mathrm{term}\} \cup \{w_2, \ldots, w_{k-1}, \mathrm{term}'\}$ are identical before and after; in particular s, w_1 keep in-degree 0 and the terminals map to themselves.

Corollary D.4 (Transcript preservation)**.** *If $c_i, e_{i-1} \notin K_r$ then the entire r-round transcript satisfies $\tau_r(\sigma_i(x)) = \tau_r(x)$: each of the four keys of Lemma D.3(iii) returns a cell that would otherwise lie in K_r, all other keys return identical hits, and misses agree, so determinism forces the transcripts to coincide.*

Theorem D.5 (Cell-count floor)**.** *If a deterministic exact-match (r, H)-probe protocol is correct with probability $\ge 2/3$ on μ_k, then $Hr \ge k - 1$. Consequently every $M \in \mathcal{M}^{\mathrm{loc}}_{L,b}$ with H exact-match heads, each addressing $\le \kappa = O(1)$ cells per lookup, correct on $\mathcal{X}^{\mathrm{ptr}}_n$ obeys*

$$T_M(x) \;\ge\; \frac{k-1}{\kappa HL} - O(1) \;=\; \frac{d_{\mathcal{R}}(x) - O(\log n)}{\kappa HL}, \qquad \forall\, k \le (n-3)/2, \tag{6}$$

with no hypothesis relating n, k, H, L, b. Randomised protocols follow by Yao; the bound transfers model→protocol by Lemma D.1, so it needs no realizability.

Proof. Suppose $Hr \le k-2$. Fix a positive-probability transcript τ with fibre $F = \{x : \tau(x) = \tau\}$. For $x \in F$ the $\le Hr \le k-2$ returned cells meet $\le k-2$ of the $k-1$ disjoint aligned pairs, so some pair is fully unrevealed; let $i(x)$ be the least such index and $\Sigma(x) = \sigma_{i(x)}(x)$. By Corollary D.4, Σ maps F into F; it is an involution (for $j < i(x)$ the pairs stay killed, and the rerouted cells at $i(x)$ are not cells of x, hence not probed, so $i(\Sigma(x)) = i(x)$); and it is μ_k-measure-preserving and flips the label (Lemma D.3(i)). Thus every fibre is exactly YES/NO-balanced, so any τ-measurable output is correct with probability exactly $1/2 < 2/3$—contradiction. The model form follows from $r = LT + O(1)$, $\le \kappa$ cells per lookup, and $d_{\mathcal{R}}(x) = k+1$ (YES) or $k + \lceil \log_2 n \rceil + 3$ (NO).

Remark D.6 (What the bound rests on)**.** *This is a bandwidth/query-count argument: $\Omega(d_{\mathcal{R}}(x))$ input cells must be read while each token reads $\le \kappa HL$ of them. It requires exact-match content addressing (Remark 3.13): an argmax-over-content head that always returns* some *cell can sweep all cells in $\lceil N/H \rceil$ rounds and evade the swap (RE-D6 below). It is order-robust but not order-free. As a* lower *bound extracted model→protocol, it is unaffected by whether fast* protocols *are model-realizable (contrast RE-D1); on shallow instances ($k - 1 = O(1)$) it is correctly vacuous. It is numerically verified end-to-end (exhaustive fibre-balance check on μ_3, μ_4; game-tree optimum tight within one probe).*

D.4 Round Elimination

Theorem D.7 (Sharp constant in the sparse-id regime, $k = O(\sqrt{n}/H)$)**.** *Under μ with chain length k, if M uses exact-match hard attention and $k \le c_0\sqrt{n}/H$ (equivalently $400\,Hr(Hr+k) \le n$ at $r = (k-2)/2$), correctness $\ge 2/3$ requires $LT \ge (k - O(1))/2$—i.e. the constant of Theorem D.5 sharpens from κH to 2.*

Proof. Fix a deterministic (r, H)-probe protocol P correct with probability $\ge 2/3$ on μ_k (randomized protocols follow by Yao); at the end $r = LT + O(1)$ (Lemma D.1). Probes are the exact-match content keys of Appendix D.3; each returns ≤ 1 cell (Lemma D.2). Write $u_k = t$, w_k a dead end under YES, and symmetrically u_k a dead end, $w_k = t$ under NO.

Frontiers and lucky hits. After round j, let the *forward frontier* a_j be the largest i such that the chain prefix cells $c_1, \dots, c_i$ are probed, and the *backward frontier* b_j the largest i such that the last i cells of the in-path into t are probed; let V_j be the revealed vertices ($|V_j| \le 2 + 2Hj$: one cell of two endpoints per probe). A probe in round $j+1$ is a *lucky hit* if its key names a vertex outside V_j other than the two frontier-extension keys (u_{a_j} for the forward probe $\mathrm{out}(u_{a_j})$, and the current backward endpoint for the backward extension).

Frontier advance. If round $j+1$ has no lucky hit, then $a_{j+1} \le a_j + 1$ and $b_{j+1} \le b_j + 1$: all H probes are issued in parallel with addresses computable from the transcript through round j, and the only non-lucky probes that touch hidden cells are the two frontier extensions, each advancing one cell.

Lucky hits are rare. Condition on any transcript τ_j with no lucky hit so far and $a_j + b_j < k$. The hidden chain/decoy vertices are conditionally a uniform injective assignment over $[n] \setminus V_j$: instances agreeing on all probed cells and differing only by a permutation of hidden names are equiprobable under μ_k and produce the same τ_j. Each probe key of round $j+1$ is a fixed name, hence is lucky with probability $\le 2k/(n - |V_j|)$ (at most $2k$ hidden chain/decoy vertices), and a union bound gives

$$\Pr[\text{some lucky hit in rounds } 1..r] \;\le\; \sum_{j=0}^{r-1} \frac{2Hk}{n - 2Hj - 2} \;\le\; \frac{2Hrk}{n - 2Hr - 2}. \tag{7}$$

Indistinguishability before the frontiers meet. Condition on a transcript τ_r with no lucky hit and $a_r + b_r \le k - 3$; write $a = a_r$, $b = b_r$, and let $\beta_j = u_{k-b-1+j}$ so the probed backward suffix is $(\beta_1, \dots, \beta_b, t)$ (chain cells under YES, decoy cells under NO). We construct a μ_k-measure-preserving involution σ that flips ξ while preserving τ_r, proving $\Pr[\xi = \text{YES} \mid \tau_r] = 1/2$. Acting

on the YES branch (NO is symmetric), σ fixes every cell except three reroutings:

$$c_{k-b}:\ (u_{k-b-1}, u_{k-b}) \mapsto (u_{k-b-1}, w_k), \qquad e_{k-1}:\ (w_{k-1}, w_k) \mapsto (w_{k-1}, u_{k-b}), \qquad e_b:\ (w_b, w_{b+1}) \mapsto (w_k, w_1), \tag{8}$$

and flips ξ: the chain becomes $s \to u_1 \to \cdots \to u_{k-b-1} \to w_k \to w_1 \to \cdots \to w_b$ (dead-end, k edges) and the decoy becomes $w_{b+1} \to \cdots \to w_{k-1} \to \beta_1 \to \cdots \to \beta_b \to t$ ($k-1$ edges), so $\sigma(x)$ is a valid NO instance. *Transcript preservation:* probed cells are forward-prefix, gap, or backward-suffix cells, none of which is rerouted—the margin $a + b \le k-3$ places every rerouted cell at least one forward extension and one backward extension beyond the frontiers. All probe answers therefore agree between x and $\sigma(x)$, except for keys naming hidden vertices $(u_{k-b-1}, w_1, w_b, w_{b+1}, w_k, w_{k-1})$, which are lucky hits, and the backward-extension key $\mathrm{in}(\beta_1)$, whose two answers differ but which—had it been issued—would have hit and extended the backward frontier beyond $b_r = b$. Determinism gives $\tau_r(\sigma(x)) = \tau_r(x)$. *Measure preservation and involution:* on each transcript fibre (a, b) is fixed, so σ acts as a fixed permutation of the $2k$ chain/decoy name slots composed with the fair-coin flip $\xi \mapsto 1 - \xi$; the uniform-injective name law, the uniform cell-array order, and the balanced coin are all invariant. The NO-branch rule reroutes the same three cells back, so $\sigma \circ \sigma = \mathrm{id}$.

Conclusion. Suppose $r < (k-2)/2$. On the no-lucky-hit event, $a_r + b_r \le 2r \le k-3$, so the output is independent of ξ and correct with probability exactly $1/2$; by (7) the lucky-hit event has probability $\le 2Hrk/(n - 2Hr - 2) \le 1/8$ under the hypothesis $400\, Hr(Hr+k) \le n$. Total success is at most $\frac{1}{2} \cdot \frac{7}{8} + \frac{1}{8} = \frac{9}{16} < \frac{2}{3}$—contradiction. Hence $r \ge (k-2)/2$, and Lemma D.1 with $d_{\mathcal{R}}(x) = k + O(\log n)$ on $\mathrm{supp}(\mu_k)$ gives the model form $LT \ge (d_{\mathcal{R}}(x) - O(\log n))/2$.

Open directions:

- **RE-D1** (sharp constant $c = 2$ at $k = \Theta(n)$): the *linear floor* at $k = \Theta(n)$ is no longer open—Theorem D.5 gives it unconditionally at constant $c = \kappa H$. What remains open is only the *constant*. For the unbounded-state protocol abstraction the sharp $c = 2$ is provably unattainable once $H \ge 3$: a seed-and-grow protocol resolves the chain in $\approx 2k/(H+2)$ rounds (and blind key enumeration in $\lceil n/H \rceil$), beating $(k-2)/2$; at $H = 2$ the bidirectional chase needs exactly $\lceil k/2 \rceil$ rounds, so $c = 2$ is tight there. But both attacks carry $\Omega(k \log n)$ out-of-order scratch state and so are *not* realizable by a model with $b = O(\log n)$, where $bL \ll k \log n$; Lemma D.1 only maps model→protocol, so these attacks cap what round elimination can prove but say nothing against Theorem D.5. Hence any model-level $c = 2$ proof at $k = \Theta(n)$ must exploit the bL state bottleneck rather than round elimination—plausibly giving $T \ge \Omega(k/(bL))$ unconditionally in H; we leave this open. (This corrects the earlier framing that the $\sqrt{n}$ limitation is a mere union-bound artifact: it is a genuine phase transition at $n \asymp Hk^2/\mathrm{polylog}$.)
- **RE-D6** (exact-match vs. argmax-over-content addressing): Definition 3.9 and Theorem D.5 both assume *exact-match* content keys. A head that instead returns $\arg\max_{\text{cell}} \langle \mathbf{q}, W_K\, \mathrm{emb}(\text{content}) \rangle$ always returns *some* cell and, with a moment-curve id embedding, can implement "nearest-id" addressing that sweeps all N cells in $\lceil N/H \rceil$ rounds—defeating both the lucky-hit calculus (Theorem D.7, whose blind-guess misses never occur for a nearest-match head) and the suffix swap (a rerouted key may return a third, already-revealed cell in $\sigma_i(x)$ without charge to Hr, so Σ leaves the fibre). Establishing any linear floor for argmax-addressing heads is open. This gap is *pre-existing*: it affects Theorem D.7 equally, and is why we state exact-match content addressing explicitly.
- **RE-D2** (soft attention): real-valued scores over all positions may leak more information per round. The gap between hard and soft attention is well-known in the expressiveness literature (Hahn, 2020): soft attention can compute functions that hard attention provably cannot (e.g., counting), and our round-elimination luckiness analysis fundamentally relies on hard attention retrieving $O(1)$ cells per probe. Extending Theorem 3.10 to soft attention requires bounding the mutual information $I(\mathrm{softmax}(\mathbf{q}^\top \mathbf{K})\, \mathbf{V};\ v_j)$ between a single softmax-weighted average and an individual cell content v_j, conditioned on the protocol transcript so far. The key difficulty is that the softmax weighting concentrates adaptively: even if the expected information per cell is $O(b/n)$, the adaptive choice of query $\mathbf{q}$ can focus weight on a single cell, potentially extracting $\Theta(b)$ bits per round rather than $O(b/n)$—matching hard attention's worst case but through a mechanism that standard round-elimination cannot

handle, since it blurs the distinction between "queried" and "unqueried" cells. A resolution likely requires either (i) an entropy-based round-elimination variant (bounding conditional entropy decay per round rather than per query) or (ii) a direct-sum argument showing that resolving k adaptive pointer hops requires $\Omega(k \cdot b)$ total information regardless of extraction mechanism.

- **RE-D4** (positional probes): the content-only key restriction is sufficient but *not necessary* for the lower bound. Under random array order, the bound holds for arbitrary positional key-bits (Remark 3.13), because content keys are already unique and position is independent of chain structure. The remaining open case is *structured* input order combined with positional keys: a pure positional-addressing sweep protocol can read the entire cell array in $\lceil (2k-1)/H \rceil$ rounds regardless of array order, saturating the bound at $H = 4$ and beating it for $H \geq 5$. Whether an alternative proof technique can establish any non-trivial lower bound for pure positional-addressing protocols (without content matching) remains open.

E Proof of Proposition 3.14 (Contentful-Prefill Robustness)

E.1 The class $\widetilde{\mathcal{M}}_{L,b}$

A model $M \in \widetilde{\mathcal{M}}_{L,b}$ satisfies all clauses of Definition 3.9 except local prefill, which is replaced by:

(1′) *Contentful prefill*: during prefill, prompt positions run the L-layer network with full (bidirectional) attention among prompt positions; causal prefill is the subcase.

Bidirectional prefill is granted to make the class *stronger*: a lower bound for $\widetilde{\mathcal{M}}_{L,b}$ holds a fortiori for standard causal decoders. Per-position cached state stays $\leq b$ bits per layer; dropping this reopens C1 (an n^2-bit per-position state stores the adjacency matrix and iterates Boolean matrix squaring).

E.2 Dependency sets and the visibility budget

Lemma E.1 (Dependency sets). *For every $M \in \widetilde{\mathcal{M}}_{L,b}$, position p (prompt or generated) and layer $\ell \in \{0, \ldots, L\}$, there is a set of cells $S_\ell(p)$ such that (i) p's layer-ℓ cached state is a function of the weights and the cells in $S_\ell(p)$; (ii) $S_0(p) = \{\mathrm{cell}(p)\}$ ($\varnothing$ for generated positions); (iii) $S_\ell(p) \subseteq S_{\ell-1}(p) \cup \bigcup_{q \in A_\ell(p)} S_{\ell-1}(q)$, where $A_\ell(p)$ is the set of $\leq \kappa H$ positions whose states p reads at layer ℓ. In particular $|S_\ell(p)| \leq (1 + \kappa H)^\ell$.*

Proof. Induction on ℓ: the layer-ℓ state of p is computed from its layer-$(\ell-1)$ state and the layer-$(\ell-1)$ states of the positions it attends to at layer ℓ; each of those is, by induction, a function of its own dependency set, and each layer adds $\leq \kappa H$ sets of size $\leq (1+\kappa H)^{\ell-1}$. The attending set $A_\ell(p)$ is input-dependent (matching is adaptive), but the bound is deterministic for every input; generated positions obey the same recurrence, so reading a generated position's state is charged the same $(1+\kappa H)^L$.

Definition E.2 (Visible set). *For a run of M on x with budget T, the* visible set Vis *is the union of $S_\ell(q)$ over all state reads performed during generation (each token contributes $\leq LH\kappa$ reads; the answer read contributes one more), and $|\mathrm{Vis}| \leq (TLH\kappa + 1)(1 + \kappa H)^L$.*

E.3 Transcript preservation and the unconditional floor

Reuse μ_k and the aligned pairs $\{c_i, e_{i-1}\}$, $i = 2..k$, with the suffix-swap involutions σ_i of Lemma D.3.

Lemma E.3 (Taint propagation). *Fix x and $i \in \{2, \ldots, k\}$, and call a position p ℓ-*tainted *if $S_\ell(p) \cap \{c_i, e_{i-1}\} \neq \varnothing$ (dependency sets evaluated on x). Then for every p that is not ℓ-tainted, p's layer-ℓ state and its layer-ℓ attention matching are identical on x and on $\sigma_i(x)$.*

Proof. Induction on ℓ. Base: layer-0 states embed raw cell contents, and σ_i changes only the contents of the two positions holding c_i, e_{i-1}, which are 0-tainted. Step: assume the claim at $\ell-1$. An untainted p has identical layer-$(\ell-1)$ state, hence issues identical queries. Keys are fixed functions of raw cell contents, and σ_i preserves the source multiset and the target multiset (Lemma D.3(iii)),

so the hit/miss pattern of every query is identical on x and $\sigma_i(x)$. If a query of p matched a position q whose state differs between the worlds, then q is $(\ell{-}1)$-tainted, so $S_\ell(p) \supseteq S_{\ell-1}(q) \ni \{c_i, e_{i-1}\}$ and p is ℓ-tainted—contradiction. Boundary-key redirection (e.g. $\mathrm{in}(u_i)$ matching c_i's holder on x but e_{i-1}'s holder on $\sigma_i(x)$) is absorbed by the same argument: both holders are tainted, and any reader of a tainted state becomes tainted.

Corollary E.4 (Transcript preservation)**.** *If $\{c_i, e_{i-1}\} \cap \mathrm{Vis} = \varnothing$ then the generation transcripts satisfy $\tau(\sigma_i(x)) = \tau(x)$: every read position is untainted, so all read states—and all matching—are world-identical; induct over the read sequence. The single Vis condition subsumes both "cell unqueried" and "boundary key unqueried" of the local-prefill argument: querying a boundary key reads the state of a holder whose dependency set contains the pair, placing it in* Vis*.*

Theorem E.5 (Cell-count floor with contentful prefill)**.** *If $M \in \widetilde{\mathcal{M}}_{L,b}$ is correct with probability $\geq 2/3$ on $\mathcal{X}_n^{\mathrm{ptr}}$, then for every x with chain length $k \leq (n{-}3)/2$,*

$$\begin{aligned} T_M(x) \;&\geq\; \frac{k-1}{\kappa H L\,(1+\kappa H)^L} - \frac{1}{\kappa H L} \\ &=\; \frac{d_{\mathcal{R}}(x) - \mathcal{O}(\log n) - (1{+}\kappa H)^L}{c_p \cdot L}, \qquad c_p = \kappa H(1{+}\kappa H)^L, \end{aligned} \tag{9}$$

with no hypothesis relating n, k, H, L, b.

Proof. Deterministic M first (randomized M follows by Yao). Suppose $|\mathrm{Vis}(x)| < k{-}1$. The $k{-}1$ aligned pairs are pairwise-disjoint two-cell sets, so some pair i^* is disjoint from $\mathrm{Vis}(x)$; by Corollary E.4, $\tau(\sigma_{i^*}(x)) = \tau(x)$, and determinism forces the same output on x and $\sigma_{i^*}(x)$. But σ_{i^*} flips the YES/NO label and preserves $\mathcal{X}_n^{\mathrm{ptr}}$ (Lemma D.3), so M is wrong on one of the two instances—contradicting correctness on $\mathcal{X}_n^{\mathrm{ptr}}$. Hence correctness requires $|\mathrm{Vis}(x)| \geq k{-}1$, and Definition E.2 gives $(T_M(x)\, LH\kappa + 1)(1{+}\kappa H)^L \geq k{-}1$, which rearranges to (9) using $d_{\mathcal{R}}(x) = k + O(\log n)$ on $\mathrm{supp}(\mu_k)$ (Remark 3.5).

E.4 The doubling construction

Theorem E.6 (Pointer doubling in $\widetilde{\mathcal{M}}_{L,b}$)**.** *There exists $M \in \widetilde{\mathcal{M}}_{L,b}$ with $b = \lceil \log_2 n \rceil + O(1)$, $H = \kappa = 1$, bidirectional prefill, correct on all of $\mathcal{X}_n^{\mathrm{ptr}}$, with*

$$T_M(x) \;\leq\; \left\lceil \frac{d_{\mathcal{R}}(x) - O(\log n)}{2^L - 1} \right\rceil + \left\lceil \frac{2^L - 1}{L} \right\rceil + O(1).$$

Proof. **Prefill.** For a position p holding cell (a, b_p) define by induction $V_\ell(p) = \mathrm{succ}_{2^\ell - 1}(b_p)$ (the $(2^\ell{-}1)$-th successor of the cell's target; $\perp$ if the chain ends earlier): $V_0(p) = b_p$, and at layer $\ell \geq 1$, p issues the content query $\mathrm{out}(V_{\ell-1}(p))$; the matched position q holds the unique cell with source $V_{\ell-1}(p)$ (out-degree ≤ 1 promise), and p reads q's layer-$(\ell{-}1)$ value, obtaining $V_\ell(p) = \mathrm{succ}_{2^{\ell-1}}(V_{\ell-1}(p)) = \mathrm{succ}_{2^\ell - 1}(b_p)$. One lookup per layer; each V_ℓ is a single vertex name ($\lceil \log_2 n \rceil$ bits), so the state budget holds.

Phase 1 (maximal jumps). The frontier f, initialized at s (given in the query). At layer ℓ of a generation token the model jumps $2^{\ell-1}$ hops by reading the layer-$(\ell{-}1)$ value of the position holding the current frontier (layer 1 reads the raw cell $\mathrm{out}(f)$); the per-layer choice is jump-or-hold, so each token advances by any value in $\{0, \ldots, 2^L{-}1\}$, chosen greedily maximal. A layer returns $\perp$ exactly when the chain ends strictly inside that jump: t has out-degree 0 in $\mathcal{X}_n^{\mathrm{ptr}}$, so any jump crossing t reads a $\perp$ value propagated during prefill—t is never skipped silently. Phase 1 uses $\leq \lceil k/(2^L{-}1) \rceil$ tokens (the frontier is monotone and advances ≥ 1 hop per token, even on the $\perp$ token) and ends with the frontier inside a window of $\leq 2^{L-1}$ vertices before the chain endpoint e.

Phase 2 (window scan). From the Phase-1 frontier, walk forward one hop per layer (raw $\mathrm{out}(\cdot)$ reads) until $\mathrm{out}(v)$ misses; the dead-end vertex $v = e$ is compared to the query name t: $e = t \Rightarrow$ YES, else NO. Cost $\leq \lceil (2^L{-}1)/L \rceil$ tokens. When a Phase-1 jump lands exactly on t, the per-layer landing comparison detects it early, so Phase 2 runs only when $\perp$ fired strictly inside a jump.

The s-chain either reaches t (YES) or dead-ends at some $z \neq t$ (NO); Phases 1–2 identify the endpoint e exactly, so the answer $[e = t]$ is correct on all of $\mathcal{X}_n^{\mathrm{ptr}}$.

Remark E.7 (Causal prefill; sharp constant)**.** *The same strategy with per-layer fallback (a failed causal lookup falls back to the $+1$ raw-cell hop) solves $\mathcal{X}_n^{\mathrm{ptr}}$ in $\leq \lceil k/L \rceil + O(1)$ tokens, so causal contentful prefill lies between the per-depth constants $1/L$ and $1/(2^L-1)$; the exact causal optimum is open. In the sparse-id regime the frontier machinery of Theorem D.7 adapts—probe count $rH \to TLH\kappa$, frontier advance $\leq 2^L-1$ per token, prefill windows of radius 2^L in place of zero head start—to sharpen the constant of Theorem E.5 from $\kappa H(1+\kappa H)^L$ toward $\approx 2^L$: if $400 \cdot TLH \cdot (TLH+k) \leq n$ then $T \geq (k - 2^{L+1} - O(1))/(2^L - 1)$, matching Theorem E.6 within $1+o(1)$. We leave the explicit constant accounting to a future version; the main text relies only on the unconditional Theorem E.5.*

F Separation Constructions (Theorem 3.15)

F.1 STCON

Fix n vertices, $m = n$ edges (pad with self-loops). $\mathcal{X}_{\mathrm{lo}}$: star ($\mathrm{dist}(s,t) = 1$, $d_{\mathcal{R}} = 2$). $\mathcal{X}_{\mathrm{mid}}$: balanced binary tree ($\mathrm{dist} = \Theta(\log n)$, $d_{\mathcal{R}} = \Theta(\log n)$). $\mathcal{X}_{\mathrm{hi}}$: path ($\mathrm{dist} = n-1$, $d_{\mathcal{R}} = \Theta(n)$).

F.2 ADD

Fix n columns. $\mathcal{X}_{\mathrm{lo}}$: all kill ($d_{\mathcal{R}} = O(1)$). $\mathcal{X}_{\mathrm{mid}}$: propagate run of length $\lceil \log_2 n \rceil$ ($d_{\mathcal{R}} = \Theta(\log n)$). $\mathcal{X}_{\mathrm{hi}}$: generate at column 0, propagate at 1..n-1 ($d_{\mathcal{R}} = \Theta(n)$).

F.3 BFE (same formula)

Fix AND-caterpillar φ_n of depth $n-1$. $\mathcal{X}_{\mathrm{lo}}$: $\alpha(\ell_1) = 0$ ($d_{\mathcal{R}} = 1$). $\mathcal{X}_{\mathrm{mid}}$: first falsifier at level $\lceil \log_2 n \rceil$ ($d_{\mathcal{R}} = \Theta(\log n)$). $\mathcal{X}_{\mathrm{hi}}$: all-true ($d_{\mathcal{R}} = n-1 = \Theta(n)$).

F.4 Theoretical Guarantee Scope

Table 1: Theoretical guarantee scope by task family. T1 (sufficiency) holds universally; T3 (necessity) applies only to STCON-type instances where the semantic function is composition-hard.

Family	T1 upper	T3 lower	$\Theta(n)$ separation	Experimental role
STCON	✓	✓[†]	✓(pointer chasing)	Two-sided bound coverage
ADD	✓	open[‡]	× (AC^0 with pos. keys)	Upper envelope; probe testbed
BFE	✓	open[‡]	× (Spira (Spira, 1971) with pos. keys)	Same-formula separation

[†]Under exact-match content keys, optionally refined by arbitrary positional encodings (RoPE, ALiBi) under randomized input order (Remark 3.13). [‡]The C3 counterexamples require positional lookup (carry-lookahead for ADD, Spira (Spira, 1971) for BFE) and are blocked under the content-only restriction for our encodings: neither column-index nor leaf-index tokens appear in the parenthesized/positional encodings used, so these lookups cannot be performed.

G Canonical Rule Systems

G.1 STCON

Input: directed graph G on $[n]$, source s, target t. **Answer**: YES/NO.

Facts: $\mathrm{Reach}_i(v)$, $\mathrm{NotReach}_i(v)$ ($i \in [0..n]$), $\mathrm{Same}_i(v)$, $\mathrm{AllSame}_i$ (balanced aggregation), YES, NO.

Axioms: $\mathrm{Reach}_0(s)$; $\mathrm{NotReach}_0(v)$ for $v \neq s$; $\mathrm{edge}(u,v)$ facts.

Rules (arity ≤ 3, generator promise in-degree ≤ 2):

- $\mathrm{Reach}_i(u),\ \mathrm{edge}(u,v) \vdash \mathrm{Reach}_{i+1}(v)$
- $\mathrm{NotReach}_i(v),\ \{\mathrm{NotReach}_i(u) : u \in N^-(v)\} \vdash \mathrm{NotReach}_{i+1}(v)$

- $\mathrm{Reach}_i(t) \vdash \mathrm{YES}$
- $\mathrm{AllSame}_i,\ \mathrm{NotReach}_i(t) \vdash \mathrm{NO}$

Ground truth: YES: $d_{\mathcal{R}} = \mathrm{dist}(s,t) + 1$; NO: $d_{\mathcal{R}} = \mathrm{ecc}(s) + \lceil \log_2 n \rceil + 3$.

G.2 ADD

Facts: $\mathrm{Carry}(i) \in \{0,1\}$, $\mathrm{Out}(i)$, $\mathrm{ColType}(i)$. **Axioms**: $\mathrm{Carry}(0) = 0$, $\mathrm{ColType}(i)$.

Rules: generate $\vdash$ Carry(i+1) $= 1$; kill $\vdash$ Carry(i+1) $= 0$; propagate + Carry(i) $= c \vdash$ Carry(i+1) $= c$; Carry(i) $= c \vdash$ Out(i) $= (a_i + b_i + c) \bmod 10$.

Success: all n+1 Out(i) facts. **Ground truth**: $d_{\mathcal{R}} = \mathrm{maxrun}_P + O(1)$.

G.3 BFE

Facts: Val(g) $\in \{0,1\}$. **Axioms**: leaf values Val(ℓ) $= \alpha(\ell)$.

Rules (short-circuit): Val(u) $= 0$, u child of AND $g \vdash$ Val(g) $= 0$; Val(u) $= 1$, u child of OR $g \vdash$ Val(g) $= 1$; both children true/false for non-short-circuit cases.

Ground truth: $d_{\mathcal{R}} =$ min evidence-DAG depth (Proposition 3.3 DP).

H Experimental Details

Table 2: Model architecture and training hyperparameters.

Parameter	Value
Architecture	Decoder-only Transformer
d_{model}	256
Attention heads	4 (head dim = 64)
FFN hidden dim	1024
Activation	GELU
Layer normalization	Pre-LN
Positional encoding	Learned
Layers L	$\{1, 2, 4, 8, 12\}$
Parameters (L=4)	∼3.2M
Optimizer	AdamW (β_1=0.9, β_2=0.999)
Learning rate	3×10^{-4}
Weight decay	0.01
Batch size	128
Training steps	12,000
Training set size	32,768 instances
LR schedule	Cosine decay to 0
Linear warmup	200 steps
Gradient clipping	1.0
Dropout	0.0
Max sequence length	Task-dependent (77–170 tokens)
Vocabulary size	Task-specific (50–200)
Accuracy threshold for k^*	99.5%
Test instances per stratum	≥500

De-contamination audit (2026-07-28). All generators are shortcut-free: STCON randomizes vertex IDs and shuffles edge lists; ADD samples digits uniformly with planted propagate runs; BFE draws fresh random formulas with prescribed depth profiles. Pre-audit data were archived and all results re-run on clean generators.

Extended training diagnostic (36k steps). The BFE L=1 extended training run (Table 3) uses the same hyperparameters as the main grid (Table 2) with `-steps 36000` instead of 12,000; the

command is reproduced in the supplementary code (`scripts/run_m1_grid.sh` documents the shared hyperparameter template).

Probe training. Linear and MLP probes are trained on frozen model hidden states at the final prompt position (the last input token before CoT generation) to predict $d(x)$ as a continuous scalar (regression with MAE loss) for 50 epochs (Adam, lr 10^{-3}, 80/20 train/test split). Gain $= (M_{\text{base}} - M_{\text{probe}})/M_{\text{base}}$, where M denotes MAE; the baseline is OLS linear regression of $d(x)$ on prompt token count.

$L{=}1$ **ceiling failure analysis.** To distinguish a genuine depth limitation from a label-imbalance artifact, we stratify eval failures by $(d(x), \text{label})$ using the eval records stored in each run's `metrics.json` (no additional training or inference; the held-out split is deterministically reconstructed from the stored seed/family/n/depths arguments).

Table 3: $L{=}1$ eval failures stratified by depth $d(x)$ and label. Failure rate rises monotonically with $d(x)$ in both families; label counts are approximately balanced within each depth stratum, indicating that the ceiling reflects depth rather than sampling bias.

STCON $n{=}8$**, filler mode** (overall acc. 95.5%, 512 eval instances)

$d(x)$	Fail / Total	Fail rate	YES fail rate / NO fail rate
2	0/128	0%	– / –
4	3–4/128	3%	– / –
6	4/128	3%	– / –
8	15–18/128	12–14%	5% / 17–18%

BFE $n{=}15$**, semantic mode** (overall acc. 86.4%, mean over 4 seeds at 12k train steps) Label ratio 39:41 overall.

$d(x)$	Fail count (12k)	Fail count (36k)	Note
1	0	0	no failures at shallowest stratum
3	19	11	extended training cuts failures nearly in half
5	32	24	partial reduction
7	29	25	deepest stratum: little reduction

Overall accuracy is 95.5% (STCON) and 86.4% (BFE, mean over 4 seeds at 12k training steps). The deepest stratum alone accounts for 58–69% of all STCON failures, and extended training on BFE reduces mid-depth failures ($d(x){=}3, 5$) substantially more than deep-stratum failures ($d(x){=}7$). The one notable label asymmetry appears at STCON's deepest stratum, where NO instances fail more often than YES (17–18% vs. 5%)—plausibly because aggregating negative (non-reachability) evidence is harder for a single layer—but this remains a secondary effect relative to the depth trend.

I Where the Measured Slopes Sit Relative to the Bounds

A slope that respects a loose floor is not evidence about the floor. Table 4 therefore reports, for every STCON $n{=}8$ configuration, the measured slope divided by each bound that applies to it (`scripts/analyze_slope_bounds.py`). Three readings follow.

The applicable necessity floor is not testable at this scale. Theorem 3.10 is stated for $\mathcal{M}^{\text{loc}}_{L,b}$, whose prefill is local. Our trained Transformers attend freely within the prompt during prefill, so the floor that applies to them is Proposition 3.14's, weaker by $(1{+}\kappa H)^L$: at $\kappa H = 4$ it is $5{\times}10^{-2}$ at $L{=}1$ and $8.5{\times}10^{-11}$ at $L{=}12$, i.e. 7.7× and $9{\times}10^{9}$ below the measured slopes. No experiment on models of this kind can put pressure on that bound; the necessity result stands or falls on its proof, not on these measurements.

The nominal local floor is crossed, and this is not a contradiction. Against the stricter $1/(\kappa HL)$—the floor for models that *are* in $\mathcal{M}^{\text{loc}}_{L,b}$—the stride-1 runs sit 1.5–37× above, but two stride runs at $L{=}1$ sit *below* it (0.163 and 0.171 vs. 0.25, 4-seed means). Both hypotheses of the

theorem fail for those runs: they use contentful prefill, and they are not correct on the instance family (L=1 caps at 95.5% accuracy), whereas the bound assumes correctness. We record the crossing because it makes concrete what the ratio columns already show—the nominal floor is not an empirical constraint on trained models—rather than because it bears on the theorem.

Stride supervision is what tests the achievability envelope. Corollary 3.7's slope $\leq 1/L$ presupposes that each generated step advances L hops; with one-hop-per-step traces the models sit 2.5–9.3× above the envelope at $L \geq 4$, which is the curriculum-anchored regime described in §4.3, not a capacity limit. Stride-s traces pack s hops into a step, and the measured slopes then track $1/s$ to within $-33\%/+91\%$ and cross to 0.41× the envelope at L=4, s=4 (and 0.17× at L=1). The envelope is therefore reachable, and reached, once the supervision provides the hops the construction assumes.

Table 4: Measured k^* slope against each bound, STCON n=8, complete-case estimator on the depth-independent window (§P) for stride 1 and on the original window for the stride runs. × columns are measured/bound, so > 1 means the measurement lies above the bound. "$1/s$" is the stride prediction $\text{slope}(s{=}1)/s$ at the same L, and "ratio" the measured value over it. $\kappa H = 4$ throughout (H=4 heads, κ=1 cell per lookup).

L	s	seeds	slope	$\frac{1}{L}$	×env	$\frac{1}{\kappa H L}$	×loc	$\frac{1}{\kappa H (1+\kappa H)^L L}$	×pre	ratio
1	1	4	0.385 ± 0.202	1.000	0.38	0.250	1.54	$5.0{\times}10^{-2}$	7.7	—
2	1	4	0.439 ± 0.181	0.500	0.88	0.125	3.51	$5.0{\times}10^{-3}$	87.7	—
4	1	4	0.621 ± 0.152	0.250	2.48	0.063	9.93	$1.0{\times}10^{-4}$	$6.2{\times}10^{3}$	—
8	1	4	0.789 ± 0.192	0.125	6.31	0.031	25.25	$8.0{\times}10^{-8}$	$9.9{\times}10^{6}$	—
12	1	4	0.777 ± 0.265	0.083	9.33	0.021	37.30	$8.5{\times}10^{-11}$	$9.1{\times}10^{9}$	—
1	2	4	0.163 ± 0.103	1.000	0.16	0.250	**0.65**	$5.0{\times}10^{-2}$	3.3	0.91
4	2	4	0.221 ± 0.114	0.250	0.89	0.063	3.54	$1.0{\times}10^{-4}$	$2.2{\times}10^{3}$	0.72
12	2	4	0.341 ± 0.044	0.083	4.09	0.021	16.38	$8.5{\times}10^{-11}$	$4.0{\times}10^{9}$	0.87
1	4	4	0.171 ± 0.042	1.000	0.17	0.250	**0.68**	$5.0{\times}10^{-2}$	3.4	1.91
4	4	4	0.103 ± 0.034	0.250	0.41	0.063	1.65	$1.0{\times}10^{-4}$	$1.0{\times}10^{3}$	0.67
12	4	4	0.149 ± 0.045	0.083	1.79	0.021	7.16	$8.5{\times}10^{-11}$	$1.7{\times}10^{9}$	0.76

J Full Multi-Seed Results

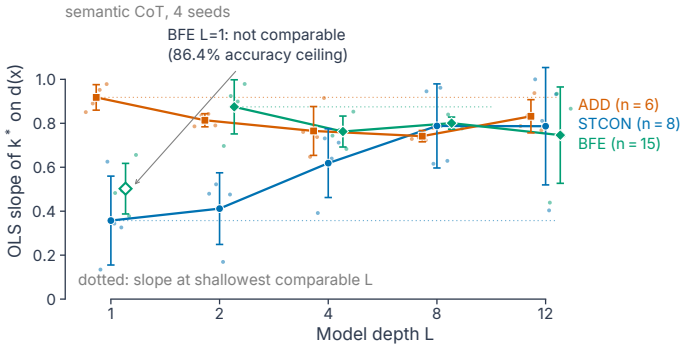


Figure 4: Slope of k^* vs. $d(x)$ as a function of model depth L (semantic mode; light dots are individual seeds, bars are mean ± 1 s.d.; values in Table 6). STCON slopes increase with L, approaching 1 as deeper models more faithfully reproduce the one-hop-per-step training signal; ADD and BFE are roughly L-independent. At $L \geq 2$, all slopes except STCON L=2 (0.41, below $1/L$=0.50) sit above the $1/L$ achievability envelope of Corollary 3.7, while all respect the $1/(\kappa HL)$ necessity floor of Theorem 3.10(i) (Table 5). BFE L=1 is capped by an accuracy ceiling (86.4%, mean over 4 seeds at 12k steps).

Trained slopes versus theoretical envelopes. Table 5 compares each semantic-mode slope against the two theoretical envelopes: the $1/L$ achievability envelope of Corollary 3.7 and the $1/(\kappa HL)$ necessity floor of Theorem 3.10(i). At $L{=}1$ every family sits within the achievability envelope; at $L \geq 2$ all slopes except STCON $L{=}2$ exceed it, while no slope falls below the necessity floor. This reveals a *curriculum–capacity factorization*: the one-hop-per-step training signal determines what each token computes (hence slopes approach 1, not 0), while model depth L governs how precisely each hop is executed—STCON slopes rise from 0.36 at $L{=}1$ to 0.79 at $L{=}12$ as deeper models implement the pointer-chase operation with less approximation error. ADD and BFE, whose per-step operations are computationally simpler, already achieve high fidelity at $L{=}1$ and thus remain roughly L-independent. The $1/L$ achievability construction (Corollary 3.7) implements L closure rounds per layer via pointer doubling—a multi-hop-per-step schedule. Approaching it would require multi-hop-per-step supervision; this is a curriculum design prediction, testable by future work on multi-round training traces. The additive $\mathcal{O}(\log n)$ slack in both bounds is large relative to $d(x) \leq 10$ at our problem sizes, so envelope comparisons at small n are indicative rather than sharp. A tight test of the $1/L$ envelope requires $d(x) \gg \log n$; the $n{=}64$ scaling runs are a first step, though the capacity limits of the 13M model prevent reaching the envelope there.

Table 5: Measured semantic-mode slopes (same data as Table 6) versus the theoretical envelopes: the $1/L$ achievability envelope (Corollary 3.7) and the $1/(\kappa HL)$ necessity floor (Theorem 3.10(i), $\kappa H = 4$). At $L \geq 2$ all trained slopes exceed the achievability envelope but respect the necessity floor; see the preceding paragraph for the small-n caveats.

Family	L	Slope (mean±std)	$1/L$ envelope	$1/(\kappa HL)$ floor
STCON $n{=}8$	1	0.357 ± 0.202	1.000	0.250
STCON $n{=}8$	2	0.412 ± 0.163	0.500	0.125
STCON $n{=}8$	4	0.619 ± 0.156	0.250	0.063
STCON $n{=}8$	8	0.788 ± 0.191	0.125	0.031
STCON $n{=}8$	12	0.787 ± 0.267	0.083	0.021
ADD $n{=}6$	1	0.918 ± 0.058	1.000	0.250
ADD $n{=}6$	2	0.814 ± 0.030	0.500	0.125
ADD $n{=}6$	4	0.765 ± 0.111	0.250	0.063
ADD $n{=}6$	8	0.741 ± 0.026	0.125	0.031
ADD $n{=}6$	12	0.832 ± 0.076	0.083	0.021
BFE $n{=}15$	1	0.503 ± 0.115	1.000	0.250
BFE $n{=}15$	2	0.875 ± 0.123	0.500	0.125
BFE $n{=}15$	4	0.762 ± 0.070	0.250	0.063
BFE $n{=}15$	8	0.801 ± 0.028	0.125	0.031
BFE $n{=}15$	12	0.746 ± 0.219	0.083	0.021

All 26 configurations show positive pooled slopes. ADD and BFE slopes are tightly concentrated (CV $\leq$ 15%) except BFE semantic $L{=}1$ (partially unsolved, overall accuracy 93.8%) and BFE semantic $L{=}12$ (one outlier seed); BFE filler $L{=}1$ is likewise partially unsolved (overall accuracy $\sim$91%) yet its slope remains tightly concentrated (CV 6.4%), confirming that d-linear scaling is robust even under underfitting. We exclude instances that never stabilize (no finite k^*) from slope calculations; these concentrate at the deeper strata (e.g., BFE $L{=}1$: 17.2% unsolved, with failure rates peaking at $d{\geq}5$). This exclusion biases slopes slightly downward because the hardest instances are removed; imputed slopes (assigning $k^* = k_{\max}$ to unsolved instances) are 5–12% steeper but noisier.

Evaluation protocol. Seed 0 evaluates 256 instances per configuration (the initial exploratory run); seeds 1–3 evaluate the full construction-time eval split (typically 512 instances). Per-stratum slopes are stable across both sizes (CV $\leq$ 20% in all cases); the larger sets marginally reduce variance. The binary-search sweep extends $k_{\text{margin}} = 4$ steps beyond the estimated k^* to confirm stability (i.e., budgets up to $k^* + 4$ are tested).

We compute STCON slopes on YES-verdict instances only (Table 6 caption), because NO instances carry a $+\lceil \log_2 n \rceil + 3$ bookkeeping offset in $d_{\mathcal{R}}$ that creates Simpson's paradox when pooled with YES instances. The YES stratum spans all four depth strata ($d \in \{2, 4, 6, 8\}$) and shows a clear increasing trend with L: slopes rise from 0.36 at $L{=}1$ to 0.79 at $L{=}12$, consistent with deeper mod-

Table 6: Multi-seed results (4 seeds each). Slope = OLS coefficient of k^* on $d_{\mathcal{R}}(x)$, computed on YES-verdict instances for STCON (see Remark 3.5). The final four rows test n-scaling: all three families remain strictly positive at doubled problem size, confirming d-linear scaling generalizes beyond the baseline n.

Family	Mode	L	Slope (mean±std)	CV (%)
STCON n=8	semantic	1	0.357 ± 0.202	56.6
STCON n=8	semantic	2	0.412 ± 0.163	39.6
STCON n=8	semantic	4	0.619 ± 0.156	25.3
STCON n=8	semantic	8	0.788 ± 0.191	24.3
STCON n=8	semantic	12	0.787 ± 0.267	33.9
ADD n=6	semantic	1	0.918 ± 0.058	6.3
ADD n=6	semantic	2	0.814 ± 0.030	3.7
ADD n=6	semantic	4	0.765 ± 0.111	14.5
ADD n=6	semantic	8	0.741 ± 0.026	3.5
ADD n=6	semantic	12	0.832 ± 0.076	9.1
BFE n=15	semantic	1	0.503 ± 0.115	22.9
BFE n=15	semantic	2	0.875 ± 0.123	14.1
BFE n=15	semantic	4	0.762 ± 0.070	9.2
BFE n=15	semantic	8	0.801 ± 0.028	3.4
BFE n=15	semantic	12	0.746 ± 0.219	29.4
STCON n=8	filler	4	0.342 ± 0.104	30.4
ADD n=6	filler	4	0.706 ± 0.108	15.3
BFE n=15	filler	1	0.689 ± 0.044	6.4
BFE n=15	filler	2	0.744 ± 0.062	8.4
BFE n=15	filler	4	0.694 ± 0.066	9.5
BFE n=15	filler	8	0.687 ± 0.098	14.3
BFE n=15	filler	12	0.606 ± 0.054	8.9
STCON n=16	semantic	4	0.552 ± 0.090	16.3
STCON n=32	semantic	4	0.231 ± 0.050	21.7
ADD n=12	semantic	4	1.000 ± 0.000	0.0
BFE n=31	semantic	4	0.909 ± 0.037	4.0

els executing the one-hop-per-step training signal more faithfully. Cross-seed CVs for STCON (25–57%) are higher than ADD and BFE, reflecting the smaller effective sample after YES filtering and per-seed variation in which depth strata contain YES instances at $d \geq 6$. BFE filler L=8 and L=12 are statistically indistinguishable (overlapping 95% CIs), consistent with absorption saturation at $L \geq 4$. The n-scaling experiment (STCON n=16, L=4 semantic) yields slope 0.552 ± 0.090—comparable to the n=8 baseline (0.619 ± 0.156)—confirming that larger instances do not admit shortcuts that erode d-linear scaling. The per-instance k^* violation rate (fraction of depth strata where the k^* curve is non-monotonic) increases from 0.09 at n=8 to 0.53 at n=16, reflecting greater measurement noise in individual instances at larger n; this noise does not contaminate the aggregate slope, which remains consistent across all 4 seeds. A further check at n=32 yields slope 0.231 ± 0.050 (CV 21.7%)—smaller than the n=16 value, but unambiguously positive, confirming that the signal does not vanish as n grows (Figure 5).

Model-scale ablation (A4). A $4\times$ larger model (d_{model}=512, H=8, L=4; 12.7M parameters vs. 3.2M baseline) trained on the same STCON n=8 semantic task has $\kappa HL = 32$ exceeding the maximum YES-instance depth ($\max d(x) = 7$), so Theorem D.5 predicts that a single thinking token suffices. Following the YES-only methodology established above, the slope is $+1.074 \pm 0.130$ over 4 seeds (per-seed: 1.041, 1.018, 1.265, 0.973; all strongly positive), with answer accuracy 99.9%. Rather than vanishing, this is the *strongest* positive slope among our STCON configurations—higher than the 3.2M baseline (0.619 ± 0.156). The budget sweep covered 200 of 2048 instances per seed (9.6%, with depth-uniform retention); the violation rate on these 200 is 0.0% (3 of 4 seeds) or 1.0% (1 seed), so first-success k^* and stable k^* coincide on 99.75% of instances. Recomputing the slope with first-success k^* over all YES instances ($\sim$1270 per seed) yields $+1.074 \pm 0.146$, reproducing the stable-k^* estimate on $10\times$ more data. The result is consistent with the curriculum–capacity factorization: the one-hop-per-step training signal anchors the slope near 1 regardless of excess throughput capacity.

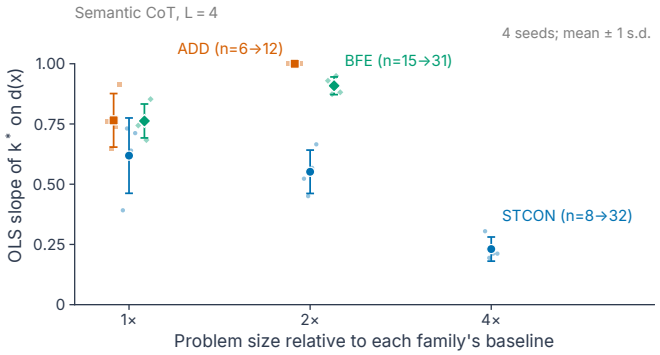


Figure 5: Slope of k^* vs. $d(x)$ as problem size grows (semantic mode, L=4). The horizontal axis is each family's problem size *relative to its own baseline*, because n counts different objects in each family (STCON vertices, ADD digits, BFE gates) and is therefore not comparable across them; absolute ranges are given in the direct labels. Every configuration stays strictly positive at every scale, which is the claim. STCON decreases with n ($0.619 \to 0.552 \to 0.231$), so no trend is fitted. Filled markers are 4-seed means with ± 1 s.d.

Scaling beyond the compact regime (n=64, planted depths). The A4 ablation eliminates the throughput constraint by enlarging the model; a complementary test *restores* the constraint by enlarging instance depth. The *same* 13M model is evaluated on STCON n=64 with planted chain depths $d \in \{32, 36, 40, 44, 48, 52, 56, 60, 64\}$, spanning the transition from $d = \kappa HL$ to $d = 2\kappa HL$. At these depths the single-token sufficiency condition of Theorem D.5 no longer holds, and the theory predicts that d-linear scaling should re-emerge.

Results (greedy decoding, 512 instances $\times$ 4 seeds):

Seed	Solved	Slope	p-value	Mean k^*
0	369/512 (72%)	0.365	1.7×10^{-11}	33.5
1	433/512 (85%)	0.651	8.2×10^{-27}	37.1
2	417/512 (81%)	0.294	1.4×10^{-5}	30.2
3	418/512 (82%)	0.719	4.3×10^{-31}	35.6
Pooled (1637 instances)		0.511	7.9×10^{-59}	34.2

The YES-only per-seed mean is 0.509 ± 0.215 (4/4 seeds positive; the table above reports all-solved instances for completeness). These slopes are measured under the depth-tied evaluation window (cap $= k^*_{\text{sem}} + 4$), which ranges from $\sim$35 to 67 across depths.

Window-sensitivity retest (uniform cap=67). The depth-tied window grants deeper instances a larger budget, which can inflate the slope if deeper instances benefit disproportionately from the wider window. To disentangle the d-linear signal from this bias, we rerun greedy evaluation on the same checkpoints with a *uniform* cap of 67 tokens—the maximum depth-tied cap—so that every instance receives identical budget headroom. Under the uniform cap, the YES-only per-seed slopes are $+0.072$ (p=0.42), $+0.239$ (p=2.6×10^{-4}), $+0.219$ (p=0.07), $+0.269$ (p=3.3×10^{-3}), with 180–201 uncensored YES instances per seed (29% censored at cap=67). The 4-seed mean is $+0.200 \pm 0.088$; a one-sample t-test against zero gives $t(3)$=4.5, p=0.01 (one-tailed). All 4 seeds are positive, but only 2 of 4 are individually significant at α=0.05—weaker than the depth-tied result where every seed achieves $p < 10^{-5}$. The slope drops 61% from the depth-tied estimate ($0.509 \to 0.200$), confirming that the original window inflated the measured effect. The d-linear signal remains positive at the aggregate level; the per-seed heterogeneity reflects that n=64 pushes the model near its accuracy frontier. Figure 6 uses the uniform-cap estimate as the authoritative n=64 data point.

A4 and planted-n=64 together show that within the *same* architecture, the d-linear slope is positive throughout the throughput transition and peaks in the most capacity-rich regime. The 13M model on n=8 yields slope $+1.074$ ($\kappa HL \gg \max d(x)$), the 3.2M baseline gives 0.619, and planted-n=64 gives 0.200 (YES-only, uniform cap=67; 0.509 under the depth-tied window). Rather than a sign

change at $d = \kappa HL$, the transition is monotonic: all three slopes are positive, and the highest occurs in the most throughput-rich regime, consistent with the curriculum anchoring the one-hop-per-step pattern. Figure 6 places all three configurations on one axis, showing that slopes decrease monotonically ($1.074 \to 0.619 \to 0.200$) as relative depth increases, with no sign change. The higher CV (44% for uniform cap, 36% for depth-tied, vs. 8–18% at $n \leq 32$) reflects that n=64 instances push the 13M model toward its accuracy frontier (80% overall vs. 99.9% at n=8); the d-linear signal remains positive in every seed under both windows.

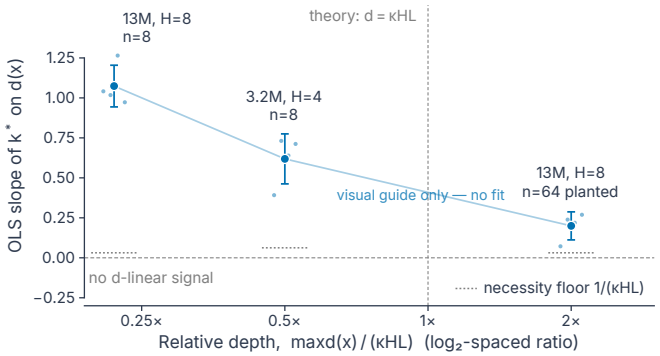


Figure 6: Curriculum–capacity factorization of the d-linear signal. Slope of k^* on $d(x)$ against instance depth measured in units of per-step throughput, $\max d(x)/\kappa HL$, for three configurations of STCON at L=4 (4 seeds each; light dots are individual seeds, bars are mean $\pm$1 s.d.). Left to right: the 13M model on n=8 (κHL=32; stable k^* from 200 of 2048 instances per seed; first-success k^* on $\sim$1270 YES instances reproduces the same slope), the 3.2M baseline on n=8 (κHL=16; 512/512 instances), and the same 13M model on n=64 with planted depths (κHL=32; uniform cap=67, 246–264 uncensored YES instances per seed). Dotted segments mark each configuration's necessity floor $1/(\kappa HL)$ from Theorem 3.10(i). All three slopes are positive and decrease monotonically ($1.074 \to 0.619 \to 0.200$) as relative depth increases; no sign change is observed. The n=64 point uses the uniform-cap estimate; the depth-tied window gives a higher value (0.509; see text for the window-sensitivity analysis).

J.1 Statistical Significance of Slopes

We assess statistical significance with one-sided t-tests (H_0: slope ≤ 0, H_1: slope > 0) and bootstrap 95% confidence intervals (10,000 resamples) for each multi-seed configuration.

J.2 Threshold and Criterion Sensitivity

The slopes in Table 6 use the per-example stable k^* (`k_star_stable_greedy`), which is threshold-independent by construction: the 99.5% accuracy threshold in §4.1 governs a different, stratum-level k^* used by the binary-search protocol (the minimal budget solving $\geq$99.5% of a depth stratum), not the slope reported there. To test robustness, we recompute the L=4 slopes under three alternative definitions across 4 seeds (Table 8). All 20 cells are strictly positive: the d-linear relationship holds for every criterion and accuracy threshold. Magnitudes differ by definition—the stratum-level $k^*(d)$ targets the budget solving $\geq \theta$ of each stratum and is therefore larger, especially in filler mode where a few hard instances dominate the tail—but the sign is invariant across definitions. Per-example stable k^* is adopted for Table 6 because it is less sensitive to within-stratum outliers; the stable-k^* column below reproduces Table 6 to within ≤ 0.001 rounding.

J.3 Probe Gain by Extraction Layer

J.4 Probe Calibration

Figure 8 plots per-instance probe predictions $\hat{d}(x)$ against ground-truth depth $d(x)$ for the three L=4 semantic configurations, using the 2-layer MLP probe of §4.5 at the best extraction layer (probe seed fixed at 1234). All three families regress toward the mean: every binned mean sits

Table 7: One-sided t-test for slope > 0 (4 seeds per configuration). All 22 configurations reject H_0 at $\alpha = 0.02$; 21/22 reject at $\alpha = 0.01$ (STCON L=1 at $p = 0.019$). 95% CI lower bounds are strictly positive in all cases.

Configuration	Slope (mean±std)	t-stat	p-value	95% CI
STCON n=8, L=1, sem	0.357 ± 0.202	3.53	0.0193	[0.035, 0.679]
STCON n=8, L=2, sem	0.412 ± 0.163	5.05	0.0075	[0.152, 0.671]
STCON n=8, L=4, sem	0.619 ± 0.156	7.91	0.0021	[0.370, 0.867]
STCON n=8, L=8, sem	0.788 ± 0.191	8.24	0.0019	[0.484, 1.092]
STCON n=8, L=12, sem	0.787 ± 0.267	5.90	0.0049	[0.362, 1.211]
STCON n=8, L=4, fill	0.342 ± 0.104	6.57	0.0036	[0.176, 0.508]
ADD n=6, L=1, sem	0.918 ± 0.058	31.64	<0.0001	[0.870, 0.965]
ADD n=6, L=2, sem	0.814 ± 0.030	54.43	<0.0001	[0.788, 0.840]
ADD n=6, L=4, sem	0.765 ± 0.111	13.76	0.0004	[0.676, 0.871]
ADD n=6, L=8, sem	0.741 ± 0.026	57.49	<0.0001	[0.726, 0.767]
ADD n=6, L=12, sem	0.832 ± 0.076	22.01	0.0001	[0.778, 0.900]
BFE n=15, L=1, sem	0.503 ± 0.115	8.73	0.0016	[0.406, 0.613]
BFE n=15, L=2, sem	0.875 ± 0.123	14.20	0.0004	[0.754, 0.959]
BFE n=15, L=4, sem	0.762 ± 0.070	21.65	0.0001	[0.705, 0.826]
BFE n=15, L=8, sem	0.801 ± 0.028	58.17	<0.0001	[0.782, 0.827]
BFE n=15, L=12, sem	0.746 ± 0.219	6.81	0.0032	[0.546, 0.900]
BFE n=15, L=4, fill	0.694 ± 0.066	21.04	0.0001	[0.640, 0.745]
STCON n=16, L=4, sem	0.552 ± 0.090	12.27	0.0004	[0.431, 0.673]
STCON n=32, L=4, sem	0.231 ± 0.050	9.21	0.0014	[0.151, 0.311]
ADD n=12, L=4, sem	1.000 ± 0.000	$>10^4$	<0.0001	[1.000, 1.000]
BFE n=31, L=4, sem	0.909 ± 0.037	49.51	<0.0001	[0.865, 0.952]
ADD n=6, L=4, fill	0.706 ± 0.108	13.06	0.0005	[0.579, 0.833]

Table 8: Threshold/criterion sensitivity of the L=4 slope (mean±std over 4 seeds). All 20 cells are positive, confirming d-linear scaling is robust to the k^* definition and accuracy threshold.

Config	stable k^*	first-success k^*	stratum θ=95%	stratum θ=99.5%
STCON n=8 sem	0.619 ± 0.156	0.418 ± 0.251	0.713 ± 0.189	0.775 ± 0.260
ADD n=6 sem	0.765 ± 0.111	0.769 ± 0.101	0.950 ± 0.100	0.950 ± 0.100
BFE n=15 sem	0.762 ± 0.070	0.716 ± 0.125	0.800 ± 0.000	0.887 ± 0.075
STCON n=8 filler	0.342 ± 0.104	0.255 ± 0.113	0.750 ± 0.204	1.950 ± 0.274
BFE n=15 filler	0.694 ± 0.066	0.291 ± 0.052	1.075 ± 0.087	1.037 ± 0.165

above the diagonal at the shallowest stratum and below it at the deepest. What separates them is the size of that bias relative to the depth spacing the probe must resolve. On STCON (MAE $= 0.72$, $R^2 = 0.81$) the largest per-stratum bias is 0.94 at $d(x)$=8, under half the depth spacing of 2, so rounding $\hat{d}(x)$ to the nearest stratum still recovers the correct one. On ADD (MAE $= 1.10$) the largest bias is 1.75 against a spacing of 1, and on BFE (MAE $= 1.45$) it is 2.22 against a spacing of 2; both exceed their spacing, so rounding lands in the wrong stratum—the mis-stratification that limits adaptive allocation on these families (§4.5). These MAE values are not directly comparable to the probe MAE reported in §4.5, which comes from a 4-layer probe trained with a mean-absolute-error objective and best-epoch checkpoint selection; Figure 7(b) quantifies the expressiveness gap between 2- and 4-layer probes.

K End-to-End Budget Allocation Results

The probe-adaptive strategy maps $\hat{d}(x) + \delta$ to the nearest valid depth in the k^* curve, where δ (depth margin) and an additional step margin are tuned via grid search on the validation split (512 instances) to maximize savings subject to ≥99.5% accuracy; all numbers in Table 9 are evaluated on the remaining 512 held-out instances. STCON achieves the best savings because its probe MAE (0.40) is small relative to the depth range ($d(x) \in \{2, 4, 6, 8\}$, spacing 2), so most instances land in the correct depth stratum. ADD's probe MAE (1.07) exceeds the depth spacing ($d(x) \in \{2, 3, 4, 5, 6\}$,

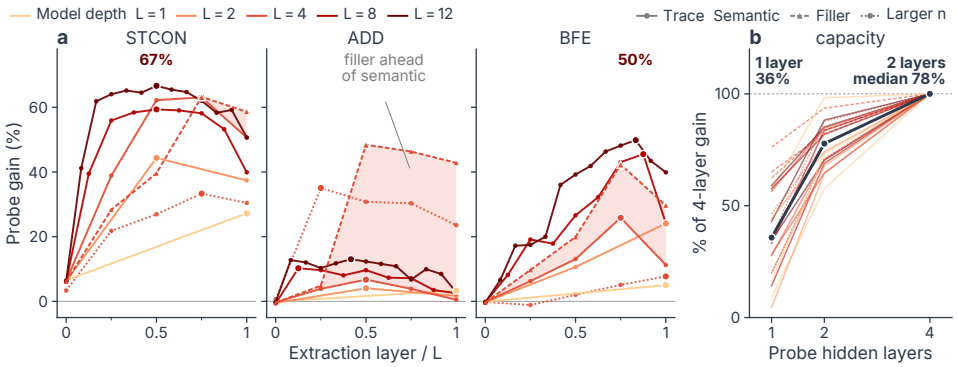


Figure 7: Depth probe gain by extraction layer (a) and probe expressiveness (b). Gain increases with model depth; a 2-layer MLP captures most of it; filler models encode $d(x)$ at least as strongly as semantic ones.

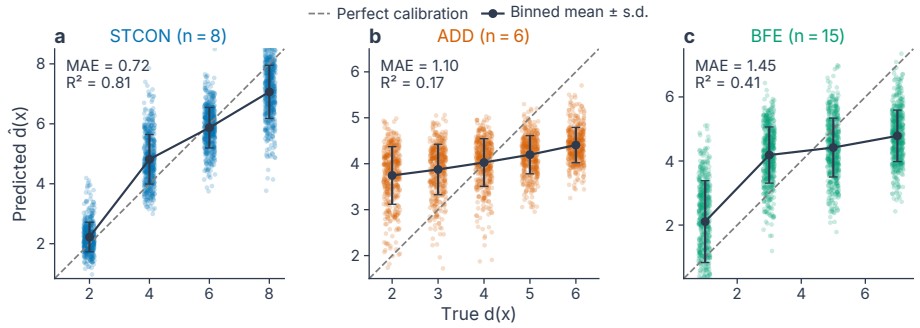


Figure 8: Probe calibration: predicted $\hat{d}(x)$ vs. true $d(x)$ for $L{=}4$ semantic models (2-layer MLP probe at the best layer, probe seed 1234; 2048 held-out instances per family). Black markers show binned mean $\pm$ s.d.; dashed line is perfect calibration. All three families regress toward the mean; only on STCON does the worst per-stratum bias (0.94) stay under half the depth spacing, which is what enables its 19.6% budget savings. ADD and BFE regress past their spacing, so rounding mis-stratifies.

spacing 1), and BFE's MAE (1.19) is large relative to its spacing ($d(x) \in \{1, 3, 5, 7\}$, spacing 2), so both mis-stratify far more often.

Why ρ alone does not order the savings. The ratio $\rho = \mathrm{MAE}/\Delta d$ is 0.20 (STCON), 1.07 (ADD) and 0.60 (BFE), which does *not* reproduce the observed ordering $+19.6\% > +5.9\% > -21.4\%$: BFE has the middling ρ and by far the worst outcome. Mis-stratification is necessary but not sufficient, because whether a mis-prediction costs steps also depends on how much slack the uniform baseline already carries. The safety margin has to fit between the deepest stratum's mean requirement $\max_s k^*(s)$ and the fixed budget k_{fixed}, so the operative quantity is that gap measured in units of probe error,

$$h \;=\; \frac{k_{\mathrm{fixed}} - \max_s k^*(s)}{\mathrm{MAE}},$$

which is 5.1 for STCON ($(5 - 2.95)/0.40$), 0.80 for ADD ($(5 - 4.14)/1.07$) and 0.40 for BFE ($(6 - 5.52)/1.19$)—the same order as the measured savings, and preserved under seed 123 (4.6, 0.78, 0.42 against $+20.0\%$, $+8.0\%$, -14.0%). Three families cannot support a functional form for h, and we claim no threshold value; the ordering is the claim, and the mechanism behind it is concrete—BFE's deepest stratum needs $k^* = 5.52$ against a uniform budget of 6, so any margin wide enough to hold 99.5% accuracy pushes the mean budget past the baseline it is meant to beat.

Seed robustness. Re-running the full pipeline with a fresh random seed (123) reproduces all qualitative findings: STCON saves $+20.0\%$ at 100% accuracy (probe MAE 0.45), ADD saves $+8.0\%$

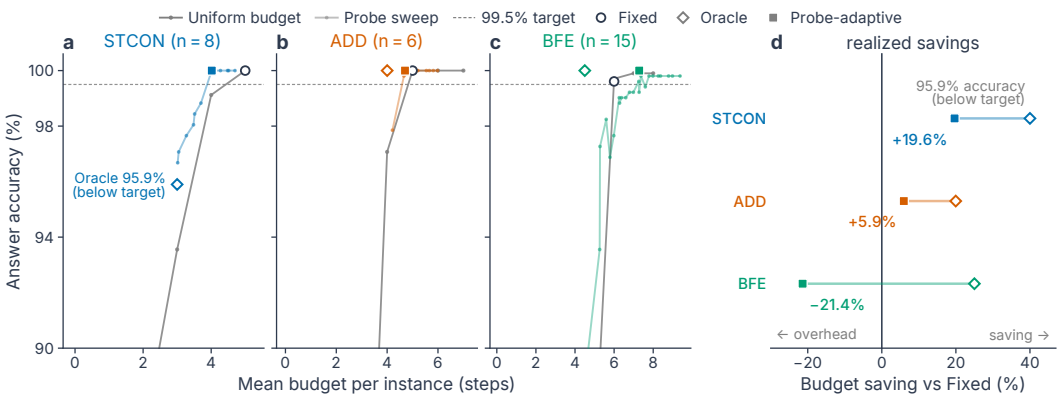


Figure 9: End-to-end budget allocation (L=4, semantic mode, 512 held-out instances; same runs as Table 9). **(a–c)** Answer accuracy against mean budget per instance. Grey traces the uniform budget, i.e. what buying more steps for *every* instance achieves; the family-coloured trace is the probe's margin sweep, i.e. the accuracy/cost frontier reachable by predicting $d(x)$. The dashed line is the 99.5% accuracy constraint against which all three strategies are defined, so a genuine saving means sitting at or above that line and to the *left* of the Fixed marker. The uniform curve continues below the plotted accuracy window. **(d)** Resulting saving relative to the uniform budget: the open diamond is the Oracle ceiling that perfect $d(x)$ knowledge would buy, the filled marker is what the probe realises. BFE's sign flip is the point: its uniform budget (k=6) already sits close to its deepest stratum's requirement (k^*=5.5), so the safety margin needed to hold 99.5% accuracy overshoots the very baseline it is trying to beat. This is why savings are not ordered by ρ alone across the three families.

Table 9: End-to-end budget allocation (L=4, semantic mode, 512 held-out instances; margins tuned on a separate 512-instance validation split). "Fixed" uses the minimum uniform budget achieving $\geq$99.5% accuracy. "Oracle" uses ground-truth $d(x)$. "Probe" uses predicted $\hat{d}(x)$ with margin tuned on the validation split to $\geq$99.5% accuracy.

Family	Strategy	Accuracy (%)	Mean budget	Total steps	Savings
STCON	Fixed (k=5)	100.0	5.00	2560	—
	Oracle	95.9	3.00	1536	40.0%
	Probe	100.0	4.02	2057	**19.6%**
ADD	Fixed (k=5)	100.0	5.00	2560	—
	Oracle	100.0	4.00	2049	20.0%
	Probe	100.0	4.70	2408	+5.9%
BFE	Fixed (k=6)	99.6	6.00	3072	—
	Oracle	100.0	4.50	2304	25.0%
	Probe	100.0	7.28	3729	−21.4%

(MAE 1.10), and BFE saves -14.0% (MAE 1.15). The seed-123 split also removes the incidental 39-instance overlap between the test and probe pools that is present under seed 42 (the two pools are disjoint under seed 123), so the savings above are unaffected by shared instances. Table 9 retains the seed-42 numbers for consistency with §4.5. Table 10 situates the depth-probe allocator among existing adaptive-budget methods. The oracle strategy's sub-100% accuracy on STCON (95.9%) arises from instances within a depth stratum that require more steps than the stratum's mean k^*—individual k^* variation is not captured by the depth-averaged map.

L Conjecture Detail: General-Transformer Lower Bound

Conjecture L.1 (General Transformers)**.** *There exists $c > 0$ such that every L-layer log-precision decoder-only Transformer of width $\mathrm{poly}(n)$ correct on $\mathcal{X}_n^{\mathrm{ptr}}$ uses at least $c \cdot \log d_{\mathcal{R}}(x)$ thinking steps on instances of depth $d_{\mathcal{R}}(x)$.*

Table 10: Comparison of adaptive CoT budget allocation approaches.

Method	Difficulty measure	Formal bound?	Scope	Mechanism
TALE (Han et al., 2025)	Surface features	×	Natural lang.	Token-level allocation
SmartThinker (He et al., 2025)	Step importance	×	Natural lang.	Step-level pruning
Early exit / aux. heads	Learned confidence	×	General	Halt prediction
Speculative decoding	Draft model agreement	×	General	Verification-based
Ours ($d(x)$ probe)	Serial depth $d(x)$	✓(T1, T3)	Synthetic	Complexity-grounded

Conjecture L.1 posits a *logarithmic* lower bound $c \cdot \log d_{\mathcal{R}}(x)$ for general poly-width Transformers, rather than the linear bound of Theorem 3.10. A poly-width model can maintain the full $n \times n$ adjacency matrix in its hidden state and perform Boolean matrix squaring at each step—resolving $\Theta(\log n)$ depth-doublings in $O(\log n)$ steps (counterexample C1). Yet even with unbounded parallelism *within* each step, the model cannot beat the information-theoretic constraint that identifying the endpoint of a k-hop chain requires reading $\Omega(\log k)$ distinct input positions: each must be attended to in at least one step, and the chain structure forces a sequential dependency among these lookups (the $(i+1)$-th pointer's identity is revealed only after the i-th is resolved through doubling). This yields a $\log k = \Theta(\log d(x))$ lower bound on serial steps. The conjecture's exponent (logarithmic) is conservative; whether a polynomial lower bound $d(x)^c$ holds for sub-polynomial-width models (e.g., width $n^{1-\varepsilon}$) is an interesting strengthening that would connect to circuit-complexity barriers.

M Counterexample Details

M.1 C1: Matrix Squaring (Poly Width)

Width $O(n^2)$ Transformer: store adjacency matrix A; each step computes $A \leftarrow A \vee (A \cdot A)$ (depth-2 MLP); after $\lceil \log_2 n \rceil$ steps, transitive closure complete. Total: $O(\log n)$ steps for any $d_{\mathcal{R}}(x)$.

M.2 C2: Contentful Prefill (constant, not order)

With prompt-to-prompt attention during prefill, position p's layer-ℓ state stores a jump pointer of span 2^ℓ (one content-addressed lookup composing two span-$2^{\ell-1}$ pointers already held in layer-$(\ell-1)$ states). After L prefill layers every position holds a span-2^L pointer, and each generated token composes jumps $2^0 + 2^1 + \cdots + 2^{L-1} = 2^L - 1$, resolving k-hop chains in $\lceil k/(2^L-1) \rceil + O_L(1)$ tokens (Theorem E.6). Two mechanisms block the naive $O(\log k)$ conclusion one might draw from pointer doubling: generation layer ℓ reads only layer-$(\ell-1)$ states, so every token restarts the layer ladder from scratch; and the write-once KV cache forbids the all-position pointer-table update that true parallel doubling would require. Contentful prefill therefore buys an exponential-in-L constant ($2^L/L$ per-depth) but *cannot break linearity*: Theorem E.5 proves the unconditional floor $T = \Omega\big(k/(\kappa H(1+\kappa H)^L L)\big)$. Local prefill is necessary for the $O(1)$-constant form of Theorem 3.10, not for the linear order (Proposition 3.14).

M.3 C3: Algebraic Shortcuts

ADD: carry c_i = value at rightmost non-propagate column $j < i$. One position-biased attention lookup per digit. $O(1)$ steps total, within $\mathcal{M}^{\text{loc}}_{L,b}$.

BFE (caterpillar): $\text{AND}(\ell_1, \text{AND}(\ell_2, \ldots)) = \bigwedge_i \ell_i$. One counting head (threshold sum at n) decides in $O(1)$ steps.

N Preliminary Experiment: Pre-trained LLMs on STCON

To test whether $d(x)$-dependent behavior exists in pre-trained language models not trained on our tasks, we evaluate GPT-2 (117M parameters) and Pythia-160M on STCON with n=20 using 3-shot CoT prompting (see supplementary code for exact prompts).

Setup. We generate 190 stratified STCON instances ($d(x) \in [2, 20]$, 10 per depth stratum) and prompt each model with natural-language graph descriptions and exemplar CoT traces. Token budgets range from 32 to 512; YES/NO answers are extracted from model outputs. These runs use uniform depth-stratified sampling without per-label balancing; the same sampling artifact identified in §N.2 could in principle affect depth-accuracy correlations. Since neither model achieves above-chance performance (below), no depth-dependent conclusion is drawn.

Results. Neither model achieves above-chance task performance:

- **GPT-2** predicts "NO" for all 190 instances regardless of token budget (overall accuracy = 35.8%, entirely from the NO-class base rate). The model generates a fixed 3-line response template without adapting to the input graph.
- **Pythia-160M** produces unparseable outputs on 59% of instances; when parseable, it shows a YES-bias (overall accuracy = 29.5%) with no meaningful correlation to $d(x)$ ($|r| < 0.03$).

Interpretation. 100M-scale pre-trained LLMs cannot perform graph connectivity reasoning from few-shot prompting—a prerequisite for testing $d(x)$-dependent behavior. The d-linear scaling phenomenon observed in §4 emerges through *task-specific training*, not from general pre-training. We address this next.

N.1 Fine-Tuned GPT-2 Exhibits d-Linear Scaling

We fine-tune GPT-2 (124M parameters; H=12 heads, L=12 layers) on STCON n=8 using the same semantic CoT traces as the 3.2M baseline.

Setup. 20,000 training examples with canonical CoT traces encode graphs and step-by-step reachability exploration as plain text using the GPT-2 tokenizer. All parameters are fine-tuned for 3 epochs (batch size 16, learning rate 5×10^{-5}, cosine schedule with 200 warmup steps) across 4 seeds. Evaluation uses the same truncated-CoT forced-answering protocol as §4: at each budget k, the model sees only the first k reasoning steps before being forced to answer.

Results. All four seeds achieve 100% valid k^* measurements (1024/1024 instances). Restricting to YES instances (where the thinking budget meaningfully constrains the model):

- **GPT-2 fine-tuned:** YES-only slope $= 0.743 \pm 0.093$ (mean $\pm$ s.d. across 4 seeds).
- **3.2M L=12 from-scratch baseline:** YES-only slope $= 0.787 \pm 0.267$ (4 seeds).
- GPT-2 slope > 0: $t = 16.1$, $p < 0.001$ (one-tailed; d-linear confirmed).
- **Equivalence:** mean difference $= -0.044$; 90% CI $[-0.35, +0.27]$ (Welch $t = -0.31$, $\mathrm{df} = 3.7$). The interval includes zero but extends beyond any pre-specified equivalence bound $\delta \leq 0.2$, so the comparison is underpowered: the data are consistent with both equal slopes and a moderate advantage for either model.

Per-depth means for YES instances confirm the monotone structure: k^* increases from 1.0 at $d(x)$=2 to 5.2 at $d(x)$=8, closely matching the 3.2M baseline trajectory.

Interpretation. Fine-tuning a pre-trained language model on STCON yields d-linear CoT scaling with a point estimate close to the from-scratch baseline (difference -0.044, 90% CI $[-0.35, +0.27]$). Despite a 39$\times$ parameter increase (124M vs. 3.2M) and qualitative architectural differences (pre-trained representations, 12 vs. 4 layers, learned positional embeddings vs. absolute), both models exhibit significantly positive d-linear slopes—consistent with the view that the scaling relationship reflects the task's serial depth structure rather than model-specific features, though the comparison is underpowered to formally establish equivalence.

N.2 A Cautionary Case: Sampling Artifacts Can Fabricate Depth-Degradation Curves

We also evaluated four instruction-tuned $\sim$7–8B models (Qwen2.5-7B-Instruct, Qwen3-8B with thinking enabled, DeepSeek-R1-Distill-Llama-8B, Mistral-7B-v0.3) on STCON (n=16) with few-shot CoT prompting, testing whether d-linear behavior extends to larger pre-trained LLMs prompted (not fine-tuned) to reason step by step. The experiment surfaced a methodological pitfall.

An initial run on Qwen2.5-7B-Instruct, sampling 204 instances uniformly at random across depths, appeared to show a textbook depth-degradation curve: accuracy fell from 100% at $d(x)$=2 to 97% at $d(x)$=5, then collapsed to 26% at $d(x)$=6 and 47% at $d(x)$=7 (overall 73%). The per-label breakdown revealed the cause: this model exhibits a strong, largely depth-independent YES-bias, and the generator's uniform sampling draws a higher proportion of NO instances at deeper strata. The apparent "depth degradation" was a re-weighting of a constant label bias against a depth-varying label ratio, not a depth effect on reasoning ability.

Re-running with $(d(x), \text{label})$-balanced stratified sampling (15 instances per depth, per label, n=16) removes the effect entirely: YES-accuracy oscillates between 67% and 100% and NO-accuracy between 13% and 47% with no monotonic trend across $d(x) \in \{1, \ldots, 15\}$. The same balanced protocol applied to the remaining three models likewise shows no consistent cross-model depth-degradation signal: computing within-label accuracy deltas from shallowest to deepest stratum across all 4 models $\times$ 2 labels (8 combinations), 4 degrade and 4 improve, with magnitudes (13–20 percentage points) all within the sampling noise expected at n=15 instances per cell. The lesson: when evaluating serial-depth effects on pre-trained LLMs with synthetic benchmarks, label ratios must be explicitly balanced across the full depth range, or a model's baseline label bias can manufacture a visually compelling but entirely spurious depth-degradation curve.

CoT scaling regression. The balanced-sampling analysis resolves the accuracy artifact but leaves the paper's central question unanswered: does CoT *length* scale with depth in pre-trained LLMs? We re-ran all four models with an enlarged, deterministic instance set (50 instances per depth $\times$ 15 depths $\times$ 2 edge-padding instances $=$ 1,150 total per model, seed 42) and measured two proxies directly via OLS regression against true instance depth $d(x)$: response character length and explicitly counted CoT steps ("Step N" markers in the trace). Table 11 reports YES-instance regressions, where genuine chain-following reasoning should manifest as a positive slope.

Table 11: OLS regression of CoT proxies on instance depth (YES instances only; STCON n=16, few-shot prompting, no fine-tuning).

Model	Proxy	Slope	R^2	p	n
Qwen3-8B	CoT steps	$+0.786$	0.542	$< 10^{-100}$	700
Qwen3-8B	Response length (chars)	$+29.3$	0.200	9×10^{-36}	700
DeepSeek-R1-8B	CoT steps	-0.011	0.001	0.52	700
DeepSeek-R1-8B	Response length (chars)	-35.1	0.018	4×10^{-4}	700
Qwen2.5-7B	CoT steps	$+0.431$	0.242	8×10^{-44}	700
Qwen2.5-7B	Response length (chars)	$+25.8$	0.182	2×10^{-32}	700
Mistral-7B	CoT steps	$+0.154$	0.054	4×10^{-10}	700
Mistral-7B	Response length (chars)	$+8.9$	0.045	2×10^{-8}	700

Instance-bootstrap 95% confidence intervals (10,000 resamples) for the step slopes are $[+0.72, +0.85]$ (Qwen3-8B), $[-0.05, +0.03]$ (DeepSeek-R1-8B), $[+0.36, +0.50]$ (Qwen2.5-7B), and $[+0.10, +0.21]$ (Mistral-7B).

Qwen3-8B (accuracy 78.7%) exhibits unambiguous d-linear CoT: each unit increase in $d(x)$ adds ≈ 0.79 reasoning steps ($R^2 = 0.54$, $p < 10^{-100}$, 95% CI $[+0.72, +0.85]$) and ≈ 29 characters. DeepSeek-R1-8B (accuracy 87.0%) shows no depth scaling in step count ($p = 0.52$) and a weakly *negative* length trend: this model employs a fixed long-think strategy (mean response $\approx 3{,}500$ characters regardless of depth), achieving higher accuracy by brute-force coverage rather than adaptive depth allocation.

Qwen2.5-7B (accuracy 52.4%) exhibits a strong YES-bias (79.1% YES-rate vs. the 60.9% base rate) and moderate d-linear signal in step count ($+0.431$ steps per unit depth, $R^2 = 0.24$, $p < 10^{-43}$). Although its accuracy is substantially lower than Qwen3-8B's, the depth-scaling slope is significantly positive—a second independent confirmation that pre-trained models can spontaneously allocate CoT effort proportional to instance depth.

Mistral-7B (accuracy 22.2%) performs at the floor, essentially guessing on a binary task. The regression detects a statistically significant but weak slope ($+0.154$, $R^2 = 0.05$, $p < 10^{-9}$); however, this

signal is unreliable given that the model is not actually solving the task—it likely reflects incidental length variation rather than genuine depth-adaptive reasoning.

The four-model comparison supports three conclusions: (i) two of four pre-trained ∼7–8B models spontaneously exhibit d-linear CoT in a few-shot setting, confirming that depth-proportional reasoning is not an artifact of training from scratch and is not unique to a single model family; (ii) higher accuracy does not imply d-linear reasoning—DeepSeek-R1-8B achieves the highest accuracy (87%) via a depth-independent fixed-budget strategy; (iii) the d-linear signal requires sufficient task competence—Mistral-7B's floor-level accuracy produces a weak statistical artifact, not genuine depth-adaptive reasoning. Figure 10 visualizes the contrast.

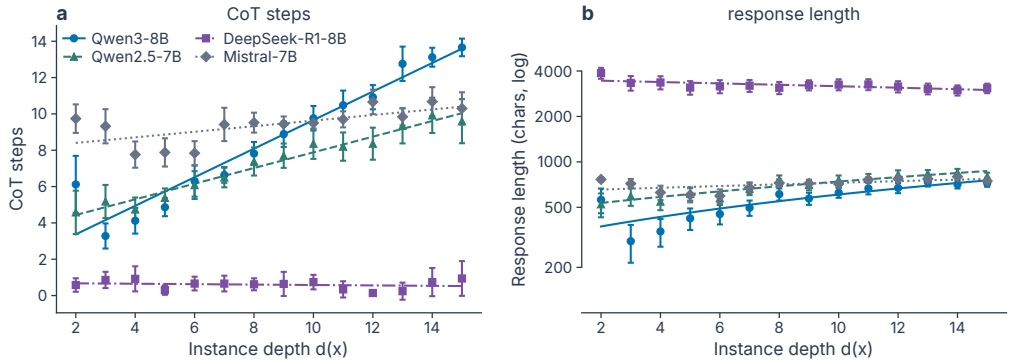


Figure 10: Few-shot CoT scaling on STCON (n=16, YES instances only, four ∼7–8B models). **(a)** CoT steps vs. instance depth: Qwen3-8B (slope = 0.79, R^2 = 0.54) and Qwen2.5-7B (slope = 0.43, R^2 = 0.24) both increase linearly; DeepSeek-R1-8B and Mistral-7B show no meaningful depth scaling. **(b)** Response length on a log axis (needed because models differ roughly tenfold in absolute length). Error bars: 95% CI of per-depth means.

O Multi-Hop Natural-Language Validation

The main-text experiments use synthetic task families where $d(x)$ is exactly computable. To test whether d-linear scaling extends toward natural-language reasoning, we evaluate a fine-tuned 7B LLM on a templated multi-hop question-answering task in which the required serial depth is known by construction.

Task and data. Each instance consists of a set of facts forming a single implication chain of length $d \in \{1, \ldots, 15\}$ together with distractor facts, a question asking for the entity reachable along the chain, and a canonical chain-of-thought that verbalizes one hop per sentence before stating the answer. The construction makes the instance depth $d(x)$ exactly the chain length: answering requires composing d relational lookups in sequence. We generate 50K training instances with depth roughly uniform in $[1, 15]$ and a disjoint held-out test set of 5000 instances. All content is template-generated; no human or web-sourced text is used.

Model and protocol. We fine-tune Qwen2.5-7B-Instruct with LoRA (r=64, α=128) on the training set, one model per seed $s \in \{42, 1, 2, 3\}$ (the seed controls both data ordering and LoRA initialization). Evaluation is greedy: each test prompt is completed with a "Reasoning:" trace and terminated at the answer or a stop marker; the measured budget proxy is the number of reasoning sentences emitted before the answer, and correctness is exact match of the final entity.

Pre-registered criteria. Before running the full evaluation we fixed the success criteria: (i) slope of sentences-vs-depth > 0 with one-sided $p < 0.01$; (ii) $R^2 > 0.5$; (iii) overall accuracy $\geq 70\%$; required to hold per seed and in aggregate.

Results. All pre-registered criteria pass for every seed and in aggregate (Table 12). The pooled regression over the 19,661 correct instances (20,000 total) gives slope = 1.0000, intercept = 0.0000,

Table 12: Multi-hop 7B validation, per seed (5000 test instances each; regression over correct instances).

Seed	Slope (sentences/depth)	R^2	p (one-sided)	Accuracy
42	1.0000	1.0000	$< 0.001^{\dagger}$	98.4%
1	1.0000	1.0000	$< 0.001^{\dagger}$	98.1%
2	1.0000	1.0000	$< 0.001^{\dagger}$	98.3%
3	1.0000	1.0000	$< 0.001^{\dagger}$	98.4%
Mean $\pm$ 95% CI	1.0000 ± 0.0000	1.0000	—	$98.3\% \pm 0.2\%$

$R^2 = 1.0000$, and overall accuracy 98.3% (per-seed range 98.1–98.4%). In fact the fit is exact: for every correct instance the number of reasoning steps equals $d(x)$, so per-depth means run $1.000, 2.000, \ldots, 15.000$ with standard deviation 0.000 at every depth, and accuracy stays uniform across depths (96.4–99.5%). †Because residuals are identically zero the t-statistic diverges and the one-sided p-value is numerically zero; we report exact equality of steps and depth rather than relying on the test statistic. Token counts also grow linearly with depth (per-seed token-vs-depth slopes 8.3–17.4), but the step count itself is seed-invariant, indicating that the models fix the number of reasoning units per instance while the surface length of each unit varies.

Scope and caveats. We scope this validation to templated language: facts and chains are synthetic, and depth is imposed by construction. Because training instances contain exactly one sentence per hop, the model learns to produce exactly $d(x)$ sentences on every instance (including incorrect ones), so the perfect slope reflects format adherence rather than an adaptive decision to reason longer on harder inputs. This demonstrates that d-linear scaling survives the transition from symbolic task families to a pre-trained 7B language model fine-tuned on a language-surface reasoning task—specifically, that the model does not *collapse* multi-hop chains into fewer reasoning steps—not a claim about unrestricted natural language or about adaptive budget allocation. The budget proxy (number of reasoning steps, counted as sentences in the trace) is coarser than the exact step counter available in the symbolic families; token counts behave analogously but with seed-dependent per-step length (see Results above).

P THE k^* MEASUREMENT WINDOW

The per-instance budget is defined as $k^*(x) = \min\{k : \text{the model is correct at every budget } k' \geq k\}$, but a sweep can only check k' up to some cap. Our sweeps used $k_{\text{cap}}(x) = \text{steps}_{\text{sem}}(x) + 4$, a function of the instance's ground-truth depth. That couples the measurement to the quantity being measured in two ways: deep instances are observed for longer, so the right-censoring rate varies systematically with $d(x)$; and, because stability must hold on all of $[k, k_{\text{cap}}]$, deep instances also face a *longer* stability test. Both effects act on the estimand, not on the noise, so no estimator computed from that data can undo them. In particular, the Kaplan–Meier restricted mean provably reduces to imputing every censored instance at $k_{\text{cap}}+1$: the cap is constant within a depth stratum, so all censoring sits at the edge of the window and there is no interior risk-set information to exploit (asserted in `tests/test_kstar_censoring.py`).

The fix is to re-measure, not to re-estimate. `eval_min_steps.py -k-cap-const` gives every instance the same window. This is well defined under forced-budget decoding: the model emits its answer when it chooses to, so raising the cap does not lengthen generations—it only adds budgets at which the unconstrained greedy answer is checked. We re-ran the sweep on the same checkpoints and the same instances with a uniform cap (16 for the $n \leq 15$ families, 20–32 for the larger ones), leaving the original measurements untouched.

Three findings. *(1) The reported slopes are a property of the models, not of the window.* Across the six ADD semantic configurations the complete-case slope is unchanged to three decimals despite the window widening from 9 to 16–20 budgets; across the five BFE n=15 configurations the largest change is 0.4%; across the five STCON n=8 configurations it is 7.1%, attained at L=1 where the model is at its accuracy ceiling (Table 13). The two configurations that do move are the ones with heavy censoring: $+4.1\%$ at n=32 and $+14.0\%$ at n=64 (13M model, 4 seeds; the main text reports

the cap=67 uniform-window estimate from a separate re-evaluation, §J). Where censoring is zero—8 of the 13 configurations, including every $L \geq 2$ run of the two families used for the depth sweep—the complete-case estimator and the censoring-aware lower bound coincide exactly, so those slopes carry no censoring bias at all. *(2) The residual censoring is model failure, not truncation.* At n=16, three windows (ground-truth-tied, uniform 24, uniform 40) give censoring $12.0\%/12.3\%/12.3\%$ and slope $0.552/0.549/0.549$: the censored instances are ones the model never gets stably right, and widening the window does not recover them. *(3) Budgets are thresholds only at small n.* The diagnostic that the wider window does expose is non-monotonicity: instances correct at some budget and wrong at a larger one. This is rare for the depth-sweep configurations (1.1–13.0% at $L \geq 2$) but common at L=1 (23–24%) and at large n (33.0% at n=32, 48.4% at n=16). Where a third of instances violate monotonicity, "the minimum budget at which the model is reliably correct" is not a well-defined instance property, and we therefore read the large-n STCON numbers only through the stratum-level accuracy-threshold estimator, which asks when a whole depth stratum reaches accuracy θ and needs no per-instance threshold to exist.

Finally, the estimator choice itself matters and is worth stating explicitly. The first-success budget $\min\{k : \text{correct at } k\}$ is not the same quantity: with a binary answer a model that guesses is correct at half of all budgets, so first-success is contaminated by the guess floor and gives slopes 2–10× smaller (e.g. 0.039 vs. 0.385 at STCON n=8, L=1). All reported slopes use the stability definition above.

Table 13: Paired re-measurement of every configuration under a depth-independent budget cap. Both columns are the complete-case stable-k^* slope—the estimator reported in the main text—computed on the same checkpoints and the same instances, differing only in the sweep window: original $k_{\text{cap}}(x) = \text{steps}_{\text{sem}}(x)+4$ (max shown as "cap$_{\text{orig}}$") vs. uniform ("cap$_{\text{unif}}$"). $\Delta\%$ is relative to the original. Only seeds present in both measurements are paired. Produced by `scripts/analyze_kstar_censoring.py -compare-file`.

		slope			censored			cap	
config	seeds	unif.	orig.	$\Delta\%$	unif.	orig.	viol.	unif.	orig.
ADD n=6, L=1	4	0.918 ± 0.058	0.918 ± 0.058	+0.0	0.000	0.000	0.000	16	9
ADD n=6, L=2	4	0.814 ± 0.030	0.814 ± 0.030	+0.0	0.000	0.000	0.009	16	9
ADD n=6, L=4	4	0.765 ± 0.111	0.765 ± 0.111	+0.0	0.000	0.000	0.000	16	9
ADD n=6, L=8	4	0.741 ± 0.026	0.741 ± 0.026	+0.0	0.000	0.000	0.001	16	9
ADD n=6, L=12	4	0.832 ± 0.076	0.832 ± 0.076	+0.0	0.001	0.001	0.001	16	9
ADD n=12, L=4	4	1.000 ± 0.000	1.000 ± 0.000	+0.0	0.000	0.000	0.184	20	16
BFE n=15, L=1	4	0.503 ± 0.115	0.503 ± 0.115	+0.0	0.134	0.134	0.231	16	11
BFE n=15, L=2	4	0.875 ± 0.123	0.875 ± 0.123	+0.0	0.000	0.000	0.122	16	11
BFE n=15, L=4	4	0.759 ± 0.076	0.762 ± 0.070	+0.4	0.000	0.000	0.039	16	11
BFE n=15, L=8	4	0.801 ± 0.028	0.801 ± 0.028	+0.0	0.000	0.000	0.011	16	11
BFE n=15, L=12	4	0.746 ± 0.219	0.746 ± 0.219	+0.0	0.000	0.000	0.039	16	11
BFE n=31, L=4	4	0.903 ± 0.044	0.909 ± 0.037	+0.6	0.000	0.000	0.116	20	13
STCON n=8, L=1	4	0.385 ± 0.202	0.357 ± 0.202	−7.1	0.034	0.039	0.241	16	11
STCON n=8, L=2	4	0.439 ± 0.181	0.412 ± 0.163	−6.1	0.001	0.001	0.130	16	11
STCON n=8, L=4	4	0.621 ± 0.152	0.619 ± 0.156	−0.3	0.000	0.000	0.117	16	11
STCON n=8, L=8	4	0.789 ± 0.192	0.788 ± 0.191	−0.1	0.000	0.000	0.121	16	11
STCON n=8, L=12	4	0.777 ± 0.265	0.787 ± 0.267	+1.2	0.000	0.000	0.011	16	11
STCON n=16, L=4	4	0.549 ± 0.089	0.552 ± 0.090	+0.6	0.123	0.120	0.484	24	19
STCON n=32, L=4	4	0.222 ± 0.017	0.231 ± 0.050	+4.1	0.222	0.149	0.330	32	27
STCON n=64, L=4 (13M)	4	0.341 ± 0.183	0.389 ± 0.014	+14.0	0.290	0.306	0.016	32	24
Filler arms (published `match` supervision; see §T):									
ADD n=6, L=4	4	0.710 ± 0.126	0.706 ± 0.108	−0.6	0.000	0.000	0.091	16	9
BFE n=15, L=4	4	0.673 ± 0.070	0.694 ± 0.066	+3.1	0.000	0.000	0.171	16	11
STCON n=8, L=4	4	0.332 ± 0.120	0.342 ± 0.104	+3.1	0.008	0.010	0.051	16	11

Q Canonicity Discussion

Different rule systems for the same task yield different depth values: for STCON, BFS-style forward reachability gives $d_{\mathcal{R}}(x) = \text{dist}_G(s,t)+1$, while a DFS-based system would give depth proportional to the DFS tree height. Constant-factor equivalence does hold for many natural pairs—bidirectional BFS gives $d_{\text{bidir}}(x) = \lceil d_{\text{BFS}}(x)/2 \rceil$, preserving the instance ordering up to a factor of two. It is

not, however, implied by polynomial size, bounded arity, soundness and completeness: those four properties admit systems whose depth is exponentially smaller.

A bounded-arity system that collapses depth. For STCON, take transitive closure,

$$\mathrm{edge}(u,v) \to \mathrm{reach}(u,v), \qquad \mathrm{reach}(u,v) \wedge \mathrm{reach}(v,w) \to \mathrm{reach}(u,w).$$

It is sound, complete, arity-2, has n^2 derivable facts (polynomial size), and addresses facts purely by *content* (vertex names), yet its serial depth is $d_{\mathrm{tc}}(x) = \lceil \log_2 \mathrm{dist}_G(s,t) \rceil + O(1)$—exponentially below d_{BFS}. The other two families admit the same phenomenon: prefix-sum carry-lookahead for ADD and Spira-style rebalancing for BFE both reach $O(\log n)$ depth with arity-2 rules. So there is no general Θ-equivalence among "reasonable" rule systems, and bounded arity is not the property that pins the canonical choice down.

Why the shortcut systems lie outside $\mathcal{M}^{\mathrm{loc}}_{L,b}$. The separating property is the *state each round requires*, not arity and not addressing mode. BFS carries an n-bit reachability frontier: round i needs only the set $\{v : \mathrm{dist}_G(s,v) \le i\}$, which fits in one bit per prompt position. Transitive closure carries a $\Theta(n^2)$-fact relation—every round's new facts $\mathrm{reach}(u,w)$ range over pairs, and each subsequent round consults them—while a model in $\mathcal{M}^{\mathrm{loc}}_{L,b}$ has $O(n \cdot bL) = O(n\mathrm{polylog}n)$ cached bits across n prompt positions, asymptotically short of n^2. Prefix-sum ADD and Spira BFE fail the same test (both require $\Theta(n)$ simultaneously-live partial aggregates per round together with unbounded fan-in over them). Note what does *not* exclude them: the content-only key restriction of Theorem 3.10 is no obstacle to transitive closure, whose lookups are exact-match content queries on vertex names. Rather, a rule system that maintains a super-linear frontier is exactly the poly-width parallel computation of C1 (§M)—transitive closure *is* matrix squaring—and is scoped out by the same width restriction that C1 identifies as the one feature breaking linearity. The canonical system is therefore depth-optimal among rule systems whose per-round frontier fits the model's cached-state budget; the honest statement of canonicity is a restriction on frontier size, and characterizing depth-optimality within that class in general remains open.

Impact on the theory. Necessity does not depend on the choice at all: Corollary 3.11 restates Theorem 3.10(i) as a bound on the instance's planted chain length, so the $\Omega(k/L)$ floor holds verbatim under transitive closure or any other sound system. A shortcut system cannot evade the floor, only mispredict it. Canonicity is used for the matching *upper* bound of T1, whose construction simulates $\mathcal{R}$ round by round, and for $d_{\mathcal{R}}$ to be efficiently computable.

Impact on empirical conclusions. The experimental claims depend on *slope* > 0 (existence of d-linear scaling), not on the specific slope value, and under any rule system with $d_{\mathcal{R}'}(x) = \Theta(d_{\mathcal{R}}(x))$ the instance ordering is preserved and slopes differ by at most a constant factor. Shortcut depth is a genuinely different prediction—$\lceil \log_2 d_{\mathcal{R}}(x) \rceil$ rather than $d_{\mathcal{R}}(x)$—and therefore directly testable against measured budgets rather than merely argued about.

Which rule system do the trained models actually follow? We regress the measured per-instance budget k^* on canonical depth $d_{\mathcal{R}}(x)$ and on two shortcut predictors—the transitive-closure depth $\lceil \log_2 \mathrm{dist}_G(s,t) \rceil + 1$ and the generic $\lceil \log_2 d_{\mathcal{R}}(x) \rceil + 1$—and compare fits by AIC (Table 14). Canonical depth wins on every configuration whose k^* is measured with less than 1% right-censoring, with median $\Delta\mathrm{AIC} \ge +8.5$ per seed against both alternatives. The comparison is reversed only on the three large-n STCON runs ($n \ge 16$), where 12–34% of instances never become stably correct at any budget and canonical $R^2 \le 0.46$—there the budget signal itself is too weak to discriminate between predictors, so those runs support neither rule system.

A second test avoids functional form entirely. If a shortcut rule system set the budget, two depth strata sharing the *same* shortcut depth would need the same k^*. For BFE, $d(x) = 5$ and $d(x) = 7$ both map to shortcut depth 4; for STCON n=8, $d(x) = 6$ and $d(x) = 8$ do. Measured budgets differ across these pairs by 0.9–2.1 steps, and the per-seed bootstrap CI excludes zero in 3–4 of 4 seeds on every configuration with $L \ge 2$ (last column of Table 14). Within a single shortcut-depth class the budget still tracks canonical depth, which is what the theory predicts and what a shortcut system cannot produce.

Table 14: Canonical vs. shortcut depth as predictors of the measured budget k^*, on the depth-independent-window re-measurement (§P), 4 seeds each. $\Delta\mathrm{AIC} > 0$ favours canonical depth; we report the median over seeds because a seed with an exact canonical fit has $\mathrm{AIC} = -\infty$ and would dominate a mean. "short" is the transitive-closure/prefix-sum depth, "log" the generic $\lceil \log_2 d \rceil + 1$. The sharp test compares two depth strata that share one shortcut depth ($d(x)$=5 vs. 7 for BFE, 6 vs. 8 for STCON): under a shortcut rule system Δk^* would be 0. CI is the mean over seeds of the per-seed bootstrap interval; "sep" counts seeds whose own CI excludes zero.

	R^2		median ΔAIC		sharp test	
config	canon	log	vs. short	vs. log	Δk^* [95% CI]	sep
BFE n=15, L=2	0.921	0.859	+255	+131	+1.73 [1.53, 1.92]	4/4
BFE n=15, L=4	0.893	0.775	+312	+207	+1.99 [1.75, 2.22]	4/4
BFE n=15, L=8	0.920	0.781	+387	+272	+2.13 [1.93, 2.32]	4/4
BFE n=15, L=12	0.839	0.741	+335	+228	+1.96 [1.73, 2.19]	4/4
STCON n=8, L=2	0.452	0.433	+27	+9	+0.89 [0.16, 1.59]	3/4
STCON n=8, L=4	0.660	0.608	+50	+22	+1.25 [0.57, 1.93]	4/4
STCON n=8, L=8	0.810	0.796	+61	+15	+1.35 [0.91, 1.82]	3/4
STCON n=8, L=12	0.838	0.813	+197	+82	+1.36 [1.05, 1.68]	4/4

R Content-Only Key Restriction: Full Argument

The content-only key restriction in Theorem 3.10 is essential to the proof technique. With positional addressing (key embeddings that depend on absolute position), a protocol can sweep the randomly-ordered cell array in $\lceil (2k-1)/H \rceil$ rounds, saturating the lower bound at $H = 4$ and beating it for $H \geq 5$. The round-elimination argument (Appendix D) requires that each probe's key be a content value—a fixed vertex name—so that hitting a hidden chain vertex is a low-probability event; positional probes bypass this because they address cells by index rather than by vertex identity.

The content-only condition is sufficient but not necessary. Under the hard distribution μ_k (which places cells in uniformly random array order), the lower bound holds even if attention keys depend on position through an arbitrary b_{pos}-bit compression $\psi\colon [n] \to \{0,1\}^{b_{\mathrm{pos}}}$—including full-resolution encodings such as RoPE or ALiBi ($b_{\mathrm{pos}} = \lceil \log_2 n \rceil$). Two properties drive this: (i) content keys are already unique (each vertex label addresses exactly one cell), so positional key-bits add no addressing precision; (ii) random array order makes cell positions statistically independent of chain structure, so positional information carries zero semantic content about which vertex is next. The actual sufficient condition for Theorem 3.10 is therefore: *either* content-only keys (for any input order), *or* randomized input order (for any key type).

S L=1 Failure Stratification

Stratifying eval failures by $(d(x), \text{label})$ using the stored eval records (Table 3) confirms that the L=1 accuracy ceiling reflects a genuine depth limitation. In both STCON (filler) and BFE (semantic), failure rates rise monotonically with $d(x)$, from 0% at the shallowest stratum to 17% (STCON) and 12–27% (BFE) at the deepest, while YES/NO failure counts remain roughly balanced within each depth stratum (BFE overall 39:41). The deepest stratum alone accounts for 58–69% of all STCON failures. Extended training (36k vs. 12k steps) on BFE reduces failures at shallow-to-mid depth ($d(x)$=3: 19→11; $d(x)$=5: 32→24) but leaves the deepest stratum largely unchanged ($d(x)$=7: 29→25)—the ceiling is structural, not an optimization or sampling artifact.

T Teacher-Forcing Discussion

Is the observed d-linear scaling a teacher-forcing artifact? We originally offered three lines of evidence that it is not. The first—the filler-token arm—does not hold up once its supervision is audited, and we report below both why and what replaced it; the equal-length padding and stride experiments stand.

Filler experiment: the published arm leaks depth through trace length. The filler experiment (§4.4) replaces every reasoning step with an identical content-free padding token, so the tokens carry no *semantic* information about the decomposition. They do carry the decomposition's *length*: the published arm emits $\mathrm{steps}_{\mathrm{sem}}(x)$ filler tokens, one per semantic step. Measuring the training splits directly (`scripts/analyze_filler_leak.py`, 32,768 instances, 4 seeds) shows how strong that channel is: the correlation between trace length and instance depth is $+1.0000$ for ADD and BFE—length *is* depth, exactly—and $+0.671$ for STCON, where length additionally carries 0.1226 of the answer's 0.8114 bits of entropy. A model trained on this supervision is told each instance's depth before it computes anything, so a depth-dependent budget measured on it cannot be attributed to the model's own discovery.

We therefore retrained the arm with the length channel cut, changing only `-filler-steps` and holding the rest of the configuration fixed (L=4, d_{model}=256, 4 heads, 32,768 instances, 12,000 steps, lr 3×10^{-4}, 4 seeds). `const:8` gives every instance exactly 8 filler tokens (one distinct length, correlation 0 by construction); `rand:8` draws the count from $U\{1,\dots,8\}$ without looking at the instance (8 distinct lengths, per-seed correlations within ±0.003, $\leq$ 0.0001 answer bits), which additionally keeps every budget in $[1,8]$ in-distribution at measurement time. Two further arms match `match`'s *average* budget while carrying no depth signal—`rand:5` on STCON and `rand:7` on BFE, means 3.00 and 3.99 against `match`'s 3.00 and 4.00—so that a collapsed slope cannot be attributed to the decoupled arms merely receiving more or less compute. ADD n=6 is retrained under `const:8` and `rand:8` as a third family.

The depth-dependence does not survive (Table 15). The slope falls from 0.332 ± 0.120 to 0.026 ± 0.121 (`const:8`) and 0.001 ± 0.006 (`rand:8`) on STCON, from 0.673 ± 0.070 to -0.033 ± 0.052 and 0.023 ± 0.026 on BFE, and from 0.710 ± 0.126 to 0.000 ± 0.000 and -0.003 ± 0.005 on ADD n=6; every decoupled arm is within one standard error of zero. This is not a training failure: every arm solves 98.5–100% of eval instances, so the models still perform the task—they simply do it within a budget that no longer grows with depth. Nor is it an artifact of how much filler each arm provides. `rand:8` sits *between* `match` and `const:8` in mean filler count (4.50 vs. 3.00 and 8.00 on STCON) yet has the *smallest* slope, so ordering the arms by compute budget does not order the effect; ordering them by $\mathrm{corr}(\text{length}, d(x))$ does. The mean-matched arms remove the budget difference altogether: at an average filler count equal to `match`'s (3.00 vs. 3.00 on STCON, 3.99 vs. 4.00 on BFE) and with no depth in the length (corr $= +0.002, +0.005$), the slope is 0.005 ± 0.009 and 0.006 ± 0.006—98.4% and 99.1% below `match`, differences of -0.326 ± 0.060 and -0.667 ± 0.035 s.e. Their violation rates (0.007, 0.012) are as low as `rand:8`'s and far below `const:8`'s (0.340, 0.561), so the collapse is also not a case of models failing the stability criterion on an unfamiliar trace length: length still varies across instances in these arms, it just no longer means anything.

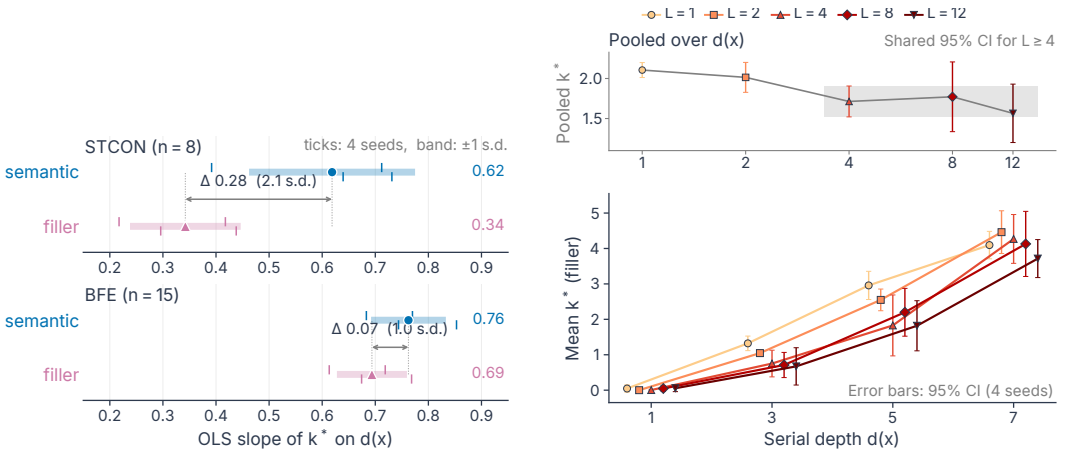
These controls identify the mechanism behind the filler-arm depth signal precisely: it is a trace-length channel carried by the supervision, not a self-discovered budget. The interpretation of the filler arm as an independent second measurement is therefore superseded by the equal-length padding and stride experiments below, which vary trace length independently of $d(x)$ and confirm that the semantic-mode d-linear law does not depend on this channel. What the arm does establish is a sharper positive: at n=8, L=4 a Transformer solves STCON at 99% with a budget flat in $d(x)$ across the tested range—any fixed average budget from 3.0 to 8.0 filler steps covers every stratum—consistent with $\lceil d(x)/L \rceil \in \{1,2\}$ over this range being small enough that per-instance depth-adaptation is not required. Depth-dependence is visible in these families only where the range of $\lceil d(x)/L \rceil$ is wide enough to require it, which is what the semantic depth sweep of §4.3 varies and what Table 4 quantifies. Figure 11a contrasts the *published* filler and semantic slopes at L=4, and Figure 11b the depth-absorption effect across model depths; both are now to be read as properties of the `match` supervision.

Equal-length padding. The equal-length padding experiment shows that models generalize to unseen prompt formats, excluding memorization of input-length/depth correlations.

Stride experiment. The stride experiment (§4.4) demonstrates that the budget scales with forward passes ($k^* \propto 1/s$), not with the trace format—models trained on stride-4 traces discover the same d-linear relationship in a 4$\times$ compressed token space.

Table 15: Filler arms with and without the trace-length channel. All arms share one training configuration and differ only in how many filler tokens each instance emits; slopes use the complete-case stable-k^* estimator on the uniform window (§P), 4 seeds. "corr" is $\mathrm{corr}(\text{trace length}, d(x))$ on the training split and "$\overline{|t|}$" the mean filler count. `match` is the published arm. The mean-matched arms (`rand:5`, `rand:7`) reproduce `match`'s average filler count with the depth signal removed. `match` is the only arm with a positive slope, and the only one whose length reveals depth.

family	arm	corr	$\overline{\lvert t \rvert}$	slope	solved	viol.
STCON n=8	`match` (published)	+0.671	3.00	$\mathbf{0.332 \pm 0.120}$	0.992	0.051
STCON n=8	`const:8`	0.000	8.00	0.026 ± 0.121	0.996	0.340
STCON n=8	`rand:8`	+0.002	4.50	0.001 ± 0.006	0.992	0.009
STCON n=8	`rand:5` (mean-matched)	+0.002	3.00	0.005 ± 0.009	0.992	0.007
BFE n=15	`match` (published)	+1.000	4.00	$\mathbf{0.673 \pm 0.070}$	1.000	0.171
BFE n=15	`const:8`	0.000	8.00	-0.033 ± 0.052	0.986	0.561
BFE n=15	`rand:8`	+0.002	4.50	0.023 ± 0.026	0.985	0.029
BFE n=15	`rand:7` (mean-matched)	+0.005	3.99	0.006 ± 0.006	0.990	0.012
ADD n=6	`match` (published)	+1.000	3.00	$\mathbf{0.710 \pm 0.126}$	1.000	0.091
ADD n=6	`const:8`	0.000	8.00	0.000 ± 0.000	1.000	0.000
ADD n=6	`rand:8`	0.000	4.50	-0.003 ± 0.005	0.999	0.001



(a) Filler vs. semantic mode (L=4). Filler slopes $\leq$ semantic.

(b) Filler mode k^* vs. $d(x)$ across model depths. Absorption saturates at $L \geq 4$.

Figure 11: Filler-token experiments: schedule constraint (a) and depth absorption (b).

Remaining caveat. The two surviving controls both hold the *format* fixed while varying depth; neither removes the possibility that a model reads its budget off some other property of the prompt rather than computing it. The filler arms above close the trace-length instance of that worry decisively—and answer it in the negative for the published arm—but they close it by making the budget flat, which shows the channel mattered without showing what the model does once the channel is gone.Settling that requires supervision under which the model chooses its own budget, i.e. variable-budget self-termination; Appendix U runs that test directly. Its result confirms the reading below: a model's self-terminated budget tracks the *length statistics of its training traces*, and efficiency pressure applied to a length-flat prior does not recover a depth-proportional budget—it collapses toward a shortcut floor, shedding accuracy on exactly the deep instances that need the computation. The depth-dependence we report should therefore be read as a property of models trained to emit a decomposition of the corresponding length, not as evidence that a Transformer discovers the decomposition unprompted.

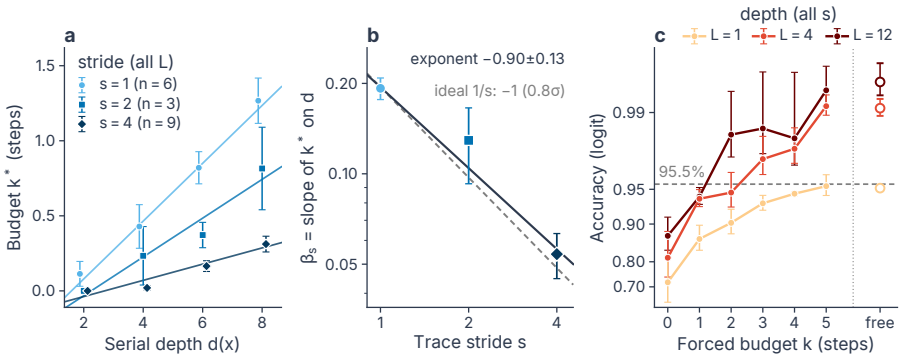


Figure 12: Stride-s experiment: $k^* \propto 1/s$ confirms steps (forward passes), not tokens, are the budget unit. L=1 single-token steps show a 95.5% accuracy ceiling regardless of budget. **(c)** pools over all strides, with 3, 9 and 6 runs at L=1, L=4 and L=12 respectively; the accuracy axis is logit-scaled and error bars are ± 1 s.e.m. across runs.

U Self-Termination Under Efficiency Pressure

The caveat of Appendix T isolates one question the format-controlled experiments cannot reach: when the model is *not* told a budget—no forced-answer position, no fixed trace length—and instead chooses for itself when to stop reasoning and emit an answer, does the budget it chooses grow with instance depth $d(x)$? We measure this under free generation: the model is prompted with the instance only, decodes greedily up to a $3\times$ cap, and the number of reasoning steps it emits before the answer token, $\text{steps}_{\text{emit}}(x)$, is its endogenous budget. We regress $\text{steps}_{\text{emit}}$ on $d(x)$ over the eval set and report the slope, R^2, and solve rate.

Why a single free-generation number is not enough. Reading $\text{steps}_{\text{emit}} \propto d(x)$ off a model as evidence of *discovery* is circular whenever the model was trained on traces whose own length grows with depth: any supervision that halts at the decomposition's closure has length $\text{steps}_{\text{sem}}(x)$, which is $d(x)$ up to a family constant (Appendix T), so a budget that reproduces it has learned to copy a length it was shown, not to compute one. The only non-circular test applies efficiency pressure to a prior whose length carries *no* depth signal, and asks whether the model then *rediscovers* a depth-proportional budget in order to stay correct while spending fewer steps. We therefore compare three supervision regimes, holding the architecture (4 layers, d_{model}=256, 4 heads), optimizer, and free-generation measurement fixed:

1. **Natural depth-proportional traces.** Supervision length $= \text{steps}_{\text{sem}}(x)$, halting at closure. This is the *inherited-proportionality* control: its free-generation budget should track $d(x)$ because the supervision already does.
2. **Constant-length flat prior.** Every trace is halt-padded to a single fixed length L (the closure step repeated to L), so trace length carries zero depth signal. Free generation should be flat in $d(x)$.
3. **Flat prior + efficiency pressure.** Starting from the regime-(2) prior, we apply REINFORCE with a correctness-gated length penalty, reward $= \mathbf{1}[\text{correct}]\,(1 - \lambda\,\text{steps}_{\text{emit}})$, an EMA baseline, and an entropy bonus. The penalty makes shorter correct answers strictly preferred, so if a depth-proportional budget is the *efficient* correct policy—spend on deep instances, save on shallow ones—this is the regime that should recover it.

Regime 3 is the crux. Its prior halts *positionally* (it was trained to emit the answer after L steps, not upon reaching closure), so a depth-proportional budget is not already in the weights; the model would have to discover content-triggered stopping from a length penalty alone. We run all three families—BFE n=15 (cleanest: $\text{steps}_{\text{sem}} = d(x)$), ADD n=6 ($\text{steps}_{\text{sem}} = d(x) - 1$, shortcuttable), STCON n=8 (branch-confounded)—at two seeds, sweeping $\lambda \in \{0.005, \ldots, 0.05\}$ for regime 3.

Result: the budget is inherited, not discovered (Table 16). Under natural traces (regime 1), free generation reproduces the supervision length exactly where that length encodes depth: BFE

and ADD give slope 1.000 at $R^2 = 1.000$ and 100% solve, with $\text{steps}_{\text{emit}} = d(x)$ and $d(x) - 1$ respectively. STCON gives only slope 0.20—not a failure of inheritance but a confirmation of it, since STCON's natural trace length is itself only weakly depth-correlated (its NO-branch closes early; $\text{corr}(\text{length}, d(x)) = 0.671$ in Appendix T), so a budget that copies the length is correctly shallow. Flattening the supervision (regime 2) flattens the budget: slope 0.000 in every family, at a constant $\text{steps}_{\text{emit}} = L$ and undiminished accuracy. The budget moves with the length statistics of the traces and nothing else.

Efficiency pressure does not restore it (regime 3). Across the entire λ sweep and both seeds the slope stays at 0.000: the model never learns to spend more on deep instances. Instead it collapses the budget to a near-constant floor ($\text{steps}_{\text{emit}} \in [0, 1]$) and absorbs the resulting error—BFE settles at the majority-class rate (0.73), ADD at 0.89 with a one-step budget. STCON makes the mechanism visible: at λ=0.01 the shallow strata keep full accuracy ($d(x) \leq 6$: ≈ 1.0) while the deep strata collapse ($d(x)$=7: 1.00 $\rightarrow$ 0.25; $d(x)$=8: $\rightarrow$ 0.45). A model that had discovered content-triggered stopping would have kept the deep budgets and lost no accuracy; instead it discards exactly the serial computation the deep instances require. This is the reward-dominant solution, not an optimization artifact: with no anchor to the depth-proportional policy, a length penalty on a shortcuttable task makes guessing within a tiny budget globally optimal, and the flat prior's positional halting leaves the policy no content-triggered stopping rule to fall back on. The identical free-generation harness recovers slope 1.000 / R^2 1.000 on regime 1, so the flat regime-3 slope is a property of the learned policy, not of the measurement.

Table 16: Self-terminated budget under three supervision regimes (free generation, 2 seeds). "slope"/"R^2" regress $\text{steps}_{\text{emit}}$ on $d(x)$; "solved" is eval accuracy; "$\overline{\text{steps}}$" the mean emitted budget. Regime 3 reports the best-accuracy λ per family; the slope is 0.000 across the full sweep $\lambda \in \{0.005, \ldots, 0.05\}$. A depth-proportional budget appears only when the training traces are depth-proportional in length (regime 1); efficiency pressure on a flat prior (regime 3) never recovers it, and collapses the budget to a shortcut floor.

family	slope	R^2	solved	$\overline{\text{steps}}$
(1) Natural depth-proportional traces (length = $\text{steps}_{\text{sem}}$, halts at closure)				
BFE n=15	**1.000**	1.000	1.000	4.50
ADD n=6	**1.000**	1.000	1.000	3.00
STCON n=8	0.20	0.18	0.94	2.00
(2) Constant-length flat prior (halt-padded to fixed L)				
BFE n=15	0.000	0.000	1.000	7.00
ADD n=6	0.000	0.000	1.000	5.00
STCON n=8	0.000	0.000	0.90	7.00
(3) Flat prior + efficiency-pressure RL (reward $\mathbf{1}[\text{correct}](1 - \lambda \text{steps})$)				
BFE n=15	0.000	0.000	0.73	0.0
ADD n=6	0.000	0.000	0.89	1.0
STCON n=8	0.000	0.000	0.89	0.0

Reading. This is the step-level counterpart of the filler control (Appendix T). There, cutting the trace-*length* channel flattened a budget that had looked depth-driven; here, a budget is depth-proportional exactly when the training traces are depth-proportional in length, and efficiency pressure from a flat prior does not manufacture the dependence from the task alone. Instance serial depth governs how much serial computation a *correct* decomposition must contain—the paper's claim—but a Transformer does not spontaneously meter its self-terminated output to that depth; it meters it to the lengths it was supervised on. Whether a different training signal—process supervision that rewards reaching closure rather than a raw length penalty, or a KL anchor to a depth-proportional reference—can induce genuine depth-adaptive self-termination is the open question this leaves.