WHY CHAIN-OF-THOUGHT HELPS: A STATISTICAL QUERY THEORY OF INTERMEDIATE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Chain-of-thought (CoT) supervision exposes intermediate variables in addition to inputs and final outputs. We develop a statistical-query (SQ) theory of when these intermediate traces make an unknown compositional computation identifiable. The target is a Boolean function computed by an unknown directed acyclic graph (DAG) whose local gates have bounded fan-in. With CoT, queries may depend on the inputs, intermediate variables, and outputs; without CoT, they may depend only on the inputs and outputs. Why can intermediate reasoning steps make a learning problem substantially easier than learning from final answers alone? We address this question in the statistical query framework. Our central observation is that chain-of-thought supervision can expose a hidden compositional structure that is statistically inaccessible at the input-output level. We model this structure as an unknown directed acyclic graph of simple local computations and show that, under natural distributional conditions, a learner can recover an equivalent computation without knowing its connections, local rules, or ordering in advance. The result clarifies when reasoning traces are informative: the relevant local configurations must occur often enough to be observed, and genuine computational dependencies must be distinguishable from spurious ones. Under these conditions, a globally difficult learning problem reduces to a collection of tractable local identification problems. For subset parity, this produces an exponential separation between learning from final answers and learning from intermediate traces, while binary addition illustrates how a long computation can be assembled from simple local steps. More broadly, our results distinguish compact representability from statistical learnability and identify the data distribution and the information exposed by the trace as central determinants of whether chain-of-thought supervision is useful.

1 INTRODUCTION

Chain-of-thought (CoT) prompting and related forms of intermediate supervision can substantially improve the performance of large language models on arithmetic, commonsense reasoning, and multi-step problem solving (Wei et al., 2022; Kojima et al., 2022; Nye et al., 2021). These empirical findings have motivated theoretical work on how intermediate computations affect expressivity, optimization, and generalization (Li et al., 2024; Yang et al., 2025; Joshi et al., 2025; Ahuja and Mansouri, 2024; Liu et al., 2023). We study a complementary statistical question: when do observed intermediate variables contain enough information to identify an unknown compositional computation?

Prior work has established that intermediate supervision can make composite tasks learnable when a suitable decomposition is supplied. Wies et al. (2023) study neural-network learning from an ordered sequence of simple subtasks. Kim and Suzuki (2025) analyze how a transformer learns parity through teacher-forced and end-to-end generation of intermediate parities. Wang et al. (2025) study permutation composition and show how suitable curricula enable efficient learning by gradient descent. These results demonstrate the power of structured intermediate computation. We build on this perspective by studying the setting in which the structure of the decomposition is itself unknown.

Statistical Identification versus Computational Expressivity. Circuit complexity asks whether a function can be represented or evaluated by a circuit of bounded size, depth, or fan-in. Statistical learning asks whether an unknown member of a function class can be identified from observations

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.

under a specified distribution. Representability and learnability are distinct axes. For example, every parity function has a linear-size fan-in-two circuit, while the class of unknown parity functions is hard for statistical-query algorithms under final-answer supervision.

The statistical query framework (Kearns, 1998; Blum et al., 1994) is well suited to isolating this distinction. The target function is represented by an unknown DAG with input nodes X, intermediate nodes Z, and output nodes Y. Every non-input node is a Boolean function of at most s parents. The parent sets, local truth tables, and topological order are unknown.

With CoT, the learner may query arbitrary bounded functions of (X, Z, Y). Without CoT, the learner may query arbitrary bounded functions of (X, Y) but may not access Z. The target family and input distribution remain fixed; only the observable variables change.

We study functional recovery: the learner seeks a compatible bounded-fan-in DAG whose intermediate and output values agree almost surely with those of the generating DAG. This formulation permits distinct DAGs that encode the same functional dependencies. When the input distribution has full support, almost-sure agreement implies pointwise agreement on every input.

Our main theorem identifies two distribution-dependent quantities controlling recovery. The parent-configuration coverage parameter γ lower-bounds the probability of every configuration of each true parent set. The functional-dependence separation parameter β requires every candidate parent set of size at most s either to determine the target node exactly or to incur prediction error at least β. Under these conditions, our constructive procedure uses at most $O\left((k+m)(s+1)2^{s+1}(d+k)^s\right)$ CoT SQs with tolerance $\tau < \min(\beta/2^{s+1}, \gamma/4)$ to recover a compatible DAG and its local truth tables. Consequently, when $s = O(1)$ and $\beta, \gamma = \Omega(1/\operatorname{poly}(d, k, m))$, both the query complexity and inverse tolerance are polynomial in the total number of nodes $N = d + k + m$.

The parameters γ and β expose statistical distinctions that circuit size and depth alone do not capture. Two functions represented by circuits of identical size and depth may have very different learning complexity because their input distributions visit the relevant local configurations with different probabilities. The coverage parameter measures the precision needed to observe each local gate configuration. The separation parameter measures the precision needed to distinguish exact functional dependencies from approximate ones. The resulting bounds connect the learnability of a compositional computation directly to the distribution of its traces.

The same viewpoint naturally incorporates imperfect trace observations. When intermediate bits pass through a known independent noise channel, the relevant local moments can be recovered by inverting that channel, with a corresponding loss in tolerance determined by its conditioning. The framework therefore relates trace quality directly to the statistical precision needed for recovery.

Subset parity provides a sharp specialization. Prefix parities form a fan-in-two trace DAG, and access to these intermediate variables permits recovery with polynomially many constant-tolerance SQs. By contrast, when queries may depend only on (X, Y), the class of unknown parities of linear sparsity requires exponentially many constant-tolerance SQs. This example isolates the effect of changing the observation model while keeping the target class fixed.

Binary addition provides a second structural example. Its partial-sum and carry variables form a bounded-fan-in DAG, placing the computation within the scope of the recovery framework whenever the coverage and separation conditions hold. It illustrates how a familiar sequential computation can be learned through statistically accessible local dependencies.

The framework also applies beyond arithmetic. In a factored finite-horizon system, each next-state variable may depend on a small state-action neighborhood; γ then measures whether observed trajectories cover the local configurations needed to identify the transition rules. In decision-tree reasoning, each node-activation variable depends on the preceding activation and the queried predicate, producing a fan-in-two trace. In propositional compositional reasoning, atomic predicates form the leaves of a parse tree, while bounded-arity operators such as negation, conjunction, and disjunction form its internal nodes. In each case, the coverage and separation parameters quantify whether the observed traces contain enough information to identify the local computation.

Contributions. Our main contributions are as follows.

- We formulate trace-supervised learning as statistical identification of an unknown bounded-fan-in DAG. The graph structure, parent sets, local truth tables, and topological order are all unknown to the learner.
- We provide a generic algorithm whose query complexity and tolerance depend explicitly on the fan-in s, the parent-configuration coverage parameter γ, and the functional-dependence separation parameter β. The procedure recovers a compatible DAG almost surely, and pointwise under full support.
- We establish an exponential separation for subset parity: final-answer SQ access requires exponentially many constant-tolerance queries at linear sparsity, whereas trace-level SQ access permits polynomial-query recovery. We also instantiate the bounded-fan-in to other arithmetic problems.
- Going beyond arithmetic circuits, we also establish a similar result for the decision-tree traversal problem. We relate imperfect intermediate supervision to recovery precision through the observation channel and discuss applications to factored transitions and bounded-arity propositional composition.

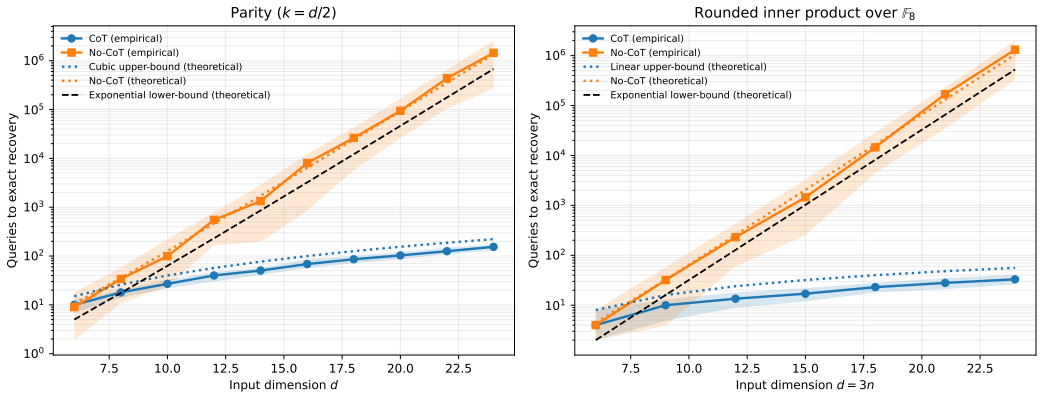

Figure 1: **SQ query complexity with and without CoT.** Results are shown for the $\mathrm{PARITY}(d, k)$ problem with $k = d/2$ **(left)** and the rounded inner-product problem over $\mathbb{F}_8$ **(right)**. Solid curves report the empirical median number of SQs required for exact recovery over 30 independently sampled targets; shaded regions show the 10th–90th percentiles. The no-CoT baseline exhaustively searches the endpoint hypothesis class using one correlation query per candidate. The dotted orange curve is its exact mean query count $(M + 1)/2$ for a uniformly random target, where M is the endpoint class size (exponential in the input dimension). The dotted blue curve is the worst-case polynomial upper-bound predicted by our theory (Corollaries 1 and 2) the dashed black curve is the exponential lower-bound $\tau^2(2M/3 - 1) \asymp M$ for any no-CoT learner using bounded endpoint-only SQs and achieving exact identification with probability at least $2/3$. All experiments use tolerance $\tau = 1/4$. Notice the excellent match between experiments and our theoretical predictions.

2 Related Work

Chain-of-thought prompting improves multi-step reasoning in both few-shot and zero-shot settings (Wei et al., 2022; Kojima et al., 2022). Closely related work on scratchpads showed that exposing intermediate computations can improve performance on algorithmic tasks such as arithmetic and program execution (Nye et al., 2021). Subsequent methods refined this idea through structured decomposition and aggregation, including least-to-most prompting, program-of-thoughts prompting, and self-consistency (Zhou et al., 2023; Chen et al., 2023). Faithfulness studies further examine the relationship between generated reasoning traces and a model's internal computation (Lanham et al., 2023). Our formalism treats intermediate computations as observed supervision and studies the statistical information they provide.

Learning from Intermediate Decompositions. The works most directly related to ours are Wies et al. (2023), Kim and Suzuki (2025), and Wang et al. (2025). Wies et al. prove neural-network

learnability results when an ordered decomposition into simple subtasks is supplied. Their result captures the broad principle that intermediate supervision can make a composite task learnable when each subtask depends on only a constant number of preceding subtask values.

Kim and Suzuki analyze gradient-based learning of parity by a transformer that recursively generates intermediate parities. They study both teacher forcing and end-to-end generation, showing how intermediate supervision and verification affect the optimization dynamics. Wang et al. study a k-fold permutation-composition task. They prove an SQ lower bound for learning from the hard-only distribution and show that transformer gradient descent succeeds under explicit or implicit easy-to-hard curricula.

These works establish qualitative and quantitative benefits of decomposing difficult tasks into simpler intermediate computations. Our analysis adds the unknown-structure dimension: the learner identifies a compatible topological order, parent sets, and local functions from trace-level statistical access. The SQ abstraction isolates the distributional conditions under which such a decomposition is identifiable, independently of the optimization procedure used to fit a particular architecture.

Expressivity, Optimization, and Generalization. Several recent works study CoT through the computational capabilities or optimization behavior of particular sequence models. Li et al. (2024) show how CoT enables transformers to perform inherently serial computations. Yang et al. (2025); Joshi et al. (2025) analyze settings in which intermediate reasoning changes the learnability of autoregressive models. Related work studies compositional and length generalization in transformers and RNNs (Ahuja and Mansouri, 2024), as well as the ability of transformers to learn automata and sequential computations (Liu et al., 2023). CoT is also connected to the in-context learning literature, where transformers have been analyzed as implementing implicit gradient-based or meta-learning procedures (von Oswald et al., 2023; Dai et al., 2022; Deutch et al., 2024).

These analyses characterize the computational capabilities, inductive biases, and optimization behavior of sequence models. The SQ perspective complements them by characterizing the information available in distributional expectations. This yields guarantees governed by trace coverage, the strength of competing functional dependencies, and the quality of the trace-observation channel.

Statistical-Query Learning. Our technical framework builds on the statistical-query model of Kearns (1998), which represents learning algorithms through approximate expectations rather than direct access to labeled samples. Classical work connects SQ learnability to Fourier structure and correlation-based complexity measures and gives hardness results for parity-like classes (Blum et al., 1994). Later work gives broader characterizations of SQ complexity and computational bounds (Feldman, 2012; Feldman and Kanade, 2012); see also the survey of Reyzin (2020).

We use this framework to compare two observation models. Under final-answer supervision, queries may depend only on (X, Y). Under trace supervision, they may also depend on the intermediate variables Z. The comparison reveals when intermediate traces contain enough statistically accessible information to recover an unknown compositional computation and how this information depends on the trace distribution and observation channel.

3 Preliminaries

3.1 The Statistical Query Model

Let D be a distribution on the input space $\{-1, 1\}^d$. A target DAG G deterministically induces intermediate variables $Z = g_G(X)$ and output variables $Y = f_G(X)$. Let $P^{\mathrm{CoT}}_{G,D}$ denote the joint distribution of (X, Z, Y), and let $P^{\mathrm{noCoT}}_{G,D}$ denote its marginal distribution on (X, Y).

Definition 1 (Statistical-Query Oracle). *Let P be a distribution on a finite domain Ω. For $\tau > 0$, the oracle $\mathrm{STAT}_P(\tau)$ accepts any query $\phi : \Omega \to [-1, 1]$ and returns a value $\widehat{\phi}$ satisfying*

$$\left|\widehat{\phi} - \mathbb{E}_{W\sim P}[\phi(W)]\right| \le \tau.$$

The oracle may return any value satisfying this inequality.

With CoT, the learner has access to $\mathrm{STAT}_{P^{\mathrm{CoT}}_{G,D}}(\tau)$ and may therefore query any bounded function of (X, Z, Y). Without CoT, the learner has access only to $\mathrm{STAT}_{P^{\mathrm{noCoT}}_{G,D}}(\tau)$ and may query bounded functions of (X, Y), but not of Z. When the underlying distribution is clear, we write simply $\mathrm{STAT}(\tau)$.

Definition 2 (Efficient SQ Learnability). *Let N denote the size parameter of a learning problem; in our DAG setting, $N = d + k + m$. A function class $\mathcal{F}$ is efficiently SQ-learnable if, for every $\varepsilon > 0$, there is an algorithm that runs in time $\mathrm{poly}(N, 1/\varepsilon)$, makes at most $\mathrm{poly}(N, 1/\varepsilon)$ statistical queries, each with tolerance at least $1/\mathrm{poly}(N, 1/\varepsilon)$, and outputs a hypothesis $\hat{f} \in \mathcal{F}$ satisfying*

$$\Pr_{X\sim D}\big(\hat{f}(X) \neq f_G(X)\big) \leq \varepsilon.$$

For a randomized learner, the running time and query complexity may additionally depend polynomially on $\log(1/\delta)$, and the guarantee must hold with probability at least $1 - \delta$ over the learner's internal randomness.

Theorem 1 establishes a stronger conclusion. Under its coverage and separation conditions, the learner uses polynomially many CoT queries of inverse-polynomial tolerance to recover a compatible DAG $\hat{G}$ satisfying

$$\Pr(\hat{V}_u \neq V_u) = 0$$

for every non-input node u. When D has full support, this equality holds for every input.

Connecting back to Classical Sample-Based Learning. It is well-known that an SQ oracle can be simulated via finite iid samples from the target distribution D, as captured in the following fact.

Fact 1 (SQ-to-Sample Simulation). *Let $W = (X, Y)$ without CoT and $W = (X, Z, Y)$ with CoT. Suppose an SQ algorithm makes at most Q adaptive queries $\phi_t : W \to [-1, 1]$, each with tolerance at least τ. Then, for every $\delta \in (0, 1)$, the algorithm can be simulated from $O\left((Q/\tau^2)\log(Q/\delta)\right)$ independent observations of W, with probability of simulation failure at most δ.*

Thus, polynomially many SQ queries of inverse-polynomial tolerance imply polynomial sample complexity under the same observation model. Without CoT, this yields ordinary PAC learning from labeled examples (X, Y). With CoT, it yields sample-efficient learning from traced examples (X, Z, Y). We refer to Feldman and Kanade (2012) for further background on SQ learning and SQ lower bounds.

3.2 Motivating Example: The Subset Parity Problem

This problem concerns learning the class $\mathrm{PARITY}(n, k)$ of functions $f_A : \{-1, 1\}^n \to \{-1, 1\}$ parameterized by subsets $A \subseteq [n]$ of size k, defined by $y = f_A(x) := \prod_{j\in A} x_j$. It is easy to see that these functions are pairwise orthogonal with respect to the uniform measure $D = \mu_n$ on the input space $\{-1, 1\}^n$. It can be shown that the SQ dimension of $\mathrm{PARITY}(n, k)$ is $\binom{n}{k}$, corresponding to the number of possible key sets S. This follows from the observation that for distinct $A, B \subseteq [n]$,

$$\langle f_A, f_B\rangle := \mathbb{E}_{x\sim D}[f_A(x)f_B(x)] = 0.$$

A well-known consequence is that $\mathrm{PARITY}(n, k)$ is not efficiently SQ-learnable for $k \gg 1$.

> **Question.** *What if, instead of bounded queries of the form $q(x, y)$, we allow bounded queries of the form $q(x, z, y)$ to an SQ oracle, where z is a ± 1 vector of length $\mathrm{poly}(n)$?*

The requirement that z be a ± 1 vector of length $\mathrm{poly}(n)$ ensures that the side information remains succinct. In particular, z may encode only polynomially many bits of auxiliary information as a function of the input size, and therefore cannot serve as an unbounded repository of external knowledge or computation.

4 Learning Juntas on a DAG with Chain-of-Thought

We consider a collection $\mathcal{F}_{d,k,m,s}$ of functions $f : \{-1, 1\}^d \to \{-1, 1\}^m$ that can be computed by a directed acyclic graph (DAG) G with input nodes $\mathcal{I} := \{x_1, \ldots, x_d\}$, intermediate nodes

$\mathcal{H} := \{z_1, \ldots, z_k\}$, and output nodes $\mathcal{O} := \{y_1, \ldots, y_m\}$. The output nodes are sinks. Every non-input node $u \in \mathcal{H} \cup \mathcal{O}$ has a parent set $\mathrm{par}_G(u)$ of size at most s and computes an unknown Boolean function of its parents. Thus, $z_i = g_i(V_{\mathrm{par}_G(z_i)})$ and $y_j = h_j(V_{\mathrm{par}_G(y_j)})$ for $i \in [k]$ and $j \in [m]$, for some functions

$$g_i : \{-1,1\}^{|\,\mathrm{par}_G(z_i)|} \to \{-1,1\}, \quad h_j : \{-1,1\}^{|\,\mathrm{par}_G(y_j)|} \to \{-1,1\}.$$

The parents of an intermediate node belong to $\mathcal{I} \cup \mathcal{H}$, subject to acyclicity, while the parents of an output node belong to $\mathcal{I} \cup \mathcal{H}$. The learner knows the node labels and the parameters d, k, m, s, but not the edges, parent sets, topological order, or local truth tables.

Writing $z = g(x)$ and $f(x) = h(x, z)$ describes the global maps induced by the DAG. The coordinates of g, however, need not be s-juntas of the original inputs: only the local function computed at each node has at most s arguments.

Remark 1. *The choice of vocabulary $\{-1, 1\}$ is purely for convenience. Analogous statements hold for any fixed finite vocabulary.*

Let $X \sim D$, and let (X, Z, Y) be the random trace obtained by evaluating G on X. For every node u, let V_u denote its random value, and for every node set S, let $V_S := (V_v)_{v \in S}$.

Definition 3. ***With CoT**, the learner may submit any query $q = \phi(X, Y, Z)$ for some function*

$$\phi : \{-1,1\}^{d+k+m} \to [-1,1].$$

The oracle $\mathrm{STAT}(\tau)$ returns a value $\hat{v}$ satisfying $|\hat{v} - \mathbb{E}[\phi(X, Z, Y)]| \le \tau$. Thus, the query may depend on any subcollection of the input, intermediate, and output variables. For example, $\phi(X, Z, Y) = X_1 Z_2 Z_3$ is a valid query.

***Without CoT**, the learner may submit any query of the form $\phi(X, Y)$ for some function $\phi : \{-1,1\}^{d+m} \to [-1,1]$, and the oracle returns a value $\hat{v}$ satisfying $|\hat{v} - \mathbb{E}[\phi(X, Y)| \le \tau$.*

4.1 The Algorithm

We now present an algorithm which simultaneously constructs a topological order $\hat{\prec}$, identifies parent sets, and recovers local truth tables. Starting from the input nodes, it repeatedly finds an intermediate node that is statistically determined by variables already recovered, adds that node to the current partial DAG, and continues until all intermediate and output nodes have been reconstructed. The result is a DAG $\hat{G}$ (alongside the topological order $\hat{\prec}$ on it) which implements exactly the same computation as the true DAG G.

Auxiliary Quantities. For every non-input node u, define its candidate-variable set by

$$\mathcal{A}_u := \begin{cases} \mathcal{I} \cup (\mathcal{H} \setminus \{u\}), & u \in \mathcal{H}, \\ \mathcal{I} \cup \mathcal{H}, & u \in \mathcal{O}, \end{cases}$$

where we recall that $\mathcal{I}$ is the collection of input nodes, $\mathcal{H}$ is the intermediate nodes, and $\mathcal{O}$ is the output nodes, with cardinality d, k, and m respectively.

For every $S \subseteq \mathcal{A}_u$ with $|S| = r \le s$, define

$$\mu^{\pm}(u, S, a) := \Pr(V_S = a, V_u = \pm 1), \tag{1}$$

$$e(u, S) := \min_{g:\{-1,1\}^r \to \{-1,1\}} \Pr(V_u \neq g(V_S)), \tag{2}$$

$$\Delta(u, S) := \sum_{a \in \{-1,1\}^r} |\mu^+(u, S, a) - \mu^-(u, S, a)|. \tag{3}$$

For each a, the smallest prediction error on the event $V_S = a$ is

$$\min(\mu^+(u, S, a), \mu^-(u, S, a)).$$

Therefore, we have the identity:

$$e(u, S) = \sum_a \min \mu^+(u, S, a), \mu^-(u, S, a) = \frac{1 - \Delta(u, S)}{2}. \tag{4}$$

In particular, $e(u, S) = 0$ iff there exists a Boolean function g such that $V_u = g(V_S)$ almost surely. Such an S need not equal the true graph-theoretic parent set of u; it is simply a set of variables that determines V_u almost surely. In particular, it always holds that $e(u, \mathrm{par}_G(u)) = 0$.

Step 1: Test Candidate Parent Sets. For every non-input node u, every candidate set $S \subseteq \mathcal{A}_u$ with $|S| = r \le s$, every $a \in \{-1,1\}^r$, and every $b \in \{-1,1\}$, submit the query

$$\phi(X, Y, Z) := \mathbf{1}\{V_S = a, V_u = b\}.$$

Let $\hat{\mu}^b(u, S, a)$ denote the oracle response. Define the following empirical quantities:

$$\hat{\Delta}(u, S) := \sum_a |\hat{\mu}^+(u, S, a) - \hat{\mu}^-(u, S, a)|, \quad \hat{p}_{u,S}(a) := \hat{\mu}^+(u, S, a) + \hat{\mu}^-(u, S, a), \tag{5}$$

and declare S viable for u iff

$$\hat{\Delta}(u, S) > 1 - \beta \quad \text{and} \quad \min_a \hat{p}_{u,S}(a) > \gamma/2, \tag{6}$$

where $\beta, \gamma \in (0, 1)$ are thresholds that will be discussed later. Here, "viable" means only that S passes the two tests in (6); it does not mean that S is necessarily the true parent set. For every viable pair (u, S), define

$$\hat{g}_{u,S}(a) := \begin{cases} +1, & \text{if } \hat{\mu}^+(u, S, a) > \hat{\mu}^-(u, S, a), \\ -1, & \text{otherwise.} \end{cases} \tag{7}$$

Step 2: Construct the DAG. Initialize $R := \mathcal{I}$ and let L be an empty ordered list. While $\mathcal{H} \setminus R \neq \varnothing$, perform the following steps:

(i) Choose any intermediate node $u \in \mathcal{H} \setminus R$ for which there exists a viable set $S \subseteq R$.

(ii) Set $\mathrm{par}_{\hat{G}}(u) := S$ and assign the gate $\hat{g}_{u,S}$ to u.

(iii) Append u to L and update $R \leftarrow R \cup \{u\}$.

After all intermediate nodes have been added, process the output nodes in any order. For each $u \in \mathcal{O}$, choose any viable set $S \subseteq R$, set $\mathrm{par}_{\hat{G}}(u) := S$, assign the gate $\hat{g}_{u,S}$ to u, and append u to L. The correctness proof shows that such a viable set always exists.

Define $\hat{\prec}$ by placing the input nodes first, followed by the non-input nodes in the order in which they appear in L. Return the learned DAG $\hat{G}$ and the topological order $\hat{\prec}$.

Inference. Given an input x, evaluate the learned DAG in the learned topological order $\hat{\prec}$. For every input node v, set $\hat{V}_v := x_v$. For every non-input node u with parent set $S_u := \mathrm{par}_{\hat{G}}(u)$, define

$$\hat{V}_u := \hat{g}_{u,S_u}(\hat{V}_{S_u}). \tag{8}$$

4.2 Correctness of the Algorithm

We impose the following conditions on the represented target DAG and the trace distribution.

Condition 1. *There exists a known $\beta \in (0, 1/2]$ such that, for every non-input node u and every candidate set $S \subseteq \mathcal{A}_u$ of size at most s, it holds that $e(u, S) = 0$ OR $e(u, S) \ge \beta$.*

Condition 2. *There exists a known $\gamma > 0$ such that, for every non-input node u and every configuration $a \in \{-1,1\}^{|\mathrm{par}_G(u)|}$, it holds that $\Pr(V_{\mathrm{par}_G(u)} = a) \ge \gamma$.*

Condition 1 separates exact determining sets from all other candidate sets. Condition 2 ensures that every configuration of each true local gate has sufficient probability mass. In particular, for the subset parity problem, both conditions are satisfied with $s = 2$, $\beta = 1/2$ and $\gamma = 1/4$.

The following is one of our main results.

Theorem 1. *Suppose Conditions 1 and 2 are met and $0 < \tau < \min\left(\beta/2^{s+1}, \gamma/4\right)$. Then, the algorithm in Section 4.1 outputs a DAG $\hat{G}$ such that $\Pr_D(\hat{V}_u \neq V_u) = 0$ for every non-input node u. The total number of CoT queries to $\mathrm{STAT}(\tau)$ is at most*

$$Q \le (k + m)(s + 1)2^{s+1}(d + k)^s. \tag{9}$$

In particular, when $s = O(1)$ and $\beta, \gamma = \Omega(1/\mathrm{poly}(d, k, m))$, the algorithm uses polynomially many queries with inverse-polynomial tolerance τ.

Finally, if the input distribution D has full support on $\{-1,1\}^d$, then the conclusion is pointwise: for every $x \in \{-1,1\}^d$ and every non-input node u, it holds that $\hat{V}_u(x) = V_u(x)$.

Concretely, if $s = 1$, the above theorem guarantees a polynomial SQ complexity $O(N^{s+1})$ where $N := d + k + m$, for recovering an equivalent computational DAG.

5 Some Examples

We now present some important corollaries which show that some well-known hard problems that are otherwise intractable can be solved efficiently with CoT. By Fact 1, it follows immediately that all of these problems are PAC-learnable with CoT side information.

5.1 Parity

Corollary 1. *Under the uniform input distribution,* $\mathrm{PARITY}(n, k)$ *is learnable using* $O(kn^2)$ *queries to a* $\mathrm{STAT}(\tau)$ *oracle with constant tolerance* $\tau = \Omega(1)$.

The proof follows from Theorem 1 once it is establishing that Conditions 1 and 2 are met with $(s, \beta, \gamma) = (2, 1/2, 1/4)$ in the case of parity, and $(3, 1/8, 1/16)$. See the appendix for details.

Our Corollary 1 above establishes an architecture-independent exponential separation between learning with and without CoT. With CoT, the intermediate parities expose an unknown fan-in-two DAG of juntas, whose structure and local gates can be recovered using $O(kn^2)$ constant-tolerance SQs. Without CoT, the $\binom{n}{k}$ possible k-subset parities remain mutually orthogonal, forcing $\Omega\left(\binom{n}{k}\right)$ constant-tolerance queries and hence $2^{\Omega(n)}$ queries when $k = \alpha n$ for fixed $\alpha \in (0, 1)$. CoT therefore reduces the intrinsic SQ complexity of parity from exponential to polynomial. Earlier work showed that particular RNNs and transformers can exploit prescribed decompositions into intermediate parities (Wies et al., 2023; Kim and Suzuki, 2025). Our result identifies the general statistical principle underlying these model-specific phenomena: intermediate reasoning exposes simple local dependencies within an otherwise globally hard function. Because the separation is established without reference to any architecture or optimization procedure, it shows that CoT's benefit is not an accident of the inductive bias of RNNs, transformers, or another particular model class.

5.2 Finite-field multiplication

We now consider an example which is relevant to cryptographic encryption. For every $n \geq 1$, there exists a set $\mathcal{M}_n \subseteq \mathbb{F}_2^{n\times n}$ with $|\mathcal{M}_n| = 2^n$ such that $M + N$ is invertible for every distinct $M, N \in \mathcal{M}_n$. Indeed, $\mathcal{M}_n$ can be identified with the finite-field extension $\mathbb{F}_{2^n}/\mathbb{F}_2$. For each $M \in \mathcal{M}_n$, define

$$f_M(a, b) := (-1)^{a^\top Mb}, \qquad a, b \in \mathbb{F}_2^n,$$

where all operations in the exponent are over $\mathbb{F}_2$. Let A, B be independent and uniform on $\mathbb{F}_2^n$. This defines the input distribution D. Consider the following CoT trace:

$$Z_{i,0} := 0, \qquad Z_{i,j} := Z_{i,j-1} + M_{ij}B_j, \tag{10}$$

for $i, j \in [n]$, followed by

$$T_i := A_i Z_{i,n}, \qquad R_0 := 0, \qquad R_i := R_{i-1} + T_i. \tag{11}$$

Then $Z_{i,n} = (MB)_i$, $R_n = X^\top MB$, and $f_M(A, B) = (-1)^{R_n}$. Thus, the computation of $f_M(B, A)$ is a Boolean DAG with $d = 2n$ input nodes $X_i := (A_i, B_i)$, $k = n^2 + 2n - 1$ intermediate nodes $Z_{i,j}$, $m = 1$ output node R_n, and fan-in at most $s = 2$.

Consider the function class $\mathcal{B}_n := \{f_M : M \in \mathcal{M}_n\}$.

Corollary 2. *For every tolerance* $\tau \in (0, 1/2)$*, the function class* $\mathcal{B}_n$ *is exactly learnable from the CoT trace using* n^2 *queries to* $\mathrm{STAT}(\tau)$.

In contrast, any adaptive, possibly randomized learner that uses arbitrary endpoint-only queries with values in $[-1, 1]$ *and exactly identifies the target with probability at least* $2/3$*, for every target and every valid* $\mathrm{STAT}(\tau)$ *oracle, must make at least*

$$Q \geq \frac{\tau^2}{2 - 2^{-n}} \left(\frac{2^{n+1}}{3} - 1\right) = \Omega(2^n)$$

i.e., and exponential number of queries.

Hence, for any fixed positive tolerance, CoT gives a polynomial-versus-exponential separation in the input dimension $d = 2n$.

5.3 Non-Arithmetic Example: Decision-Tree Traversal

Let $\mathrm{TREEPATH}(d, h)$ denote the following problem. Let T be a complete binary decision tree of depth h, where every internal vertex v queries an input coordinate $X_{q(v)} \in \{0, 1\}$, and no coordinate is queried twice on any root-to-leaf path. The input is distributed uniformly on $\{0, 1\}^d$.

For every nonroot vertex v, let $V_v = 1$ iff the execution path on input X reaches v. The activations of the nonroot internal vertices form the CoT variables, and the leaf activations form the output. Thus, the output is the one-hot encoding of the leaf reached by the decision tree. The parent relations and queried coordinates are not given to the learner.

Corollary 3. *With CoT,* $\mathrm{TREEPATH}(d, h)$ *is SQ-learnable using* $O\left(2^h(d + 2^h)^2\right)$ *queries to* $\mathrm{STAT}(\tau)$ *with tolerance* $\tau = 2^{-(3h+4)}$. *The learned DAG computes every intermediate activation and output correctly on every input. In particular, if* $h = O(\log d)$, *then both the number of queries and the inverse tolerance are polynomial in* d.

The proof (provided in appendix) relies on showing that Conditions 1 and 2 are met with $\beta = 2^{-3h}$ and $\gamma = 2^{-h}$.

6 Concluding Remarks

This work develops a learning-theoretic account of why chain-of-thought (CoT) can dramatically improve learnability. Our central message is that the benefit of CoT is not merely heuristic or empirical: in the statistical query (SQ) framework, CoT can reduce the complexity of learning by exposing intermediate variables that decompose a difficult global computation into a directed acyclic graph (DAG) of small local circuits. When these local circuits are bounded-size juntas, the resulting problem becomes efficiently learnable even when the original input-output mapping is classically hard in the SQ sense.

This perspective yields both conceptual and technical consequences. Conceptually, it suggests that the power of CoT comes from revealing a useful computational factorization of the target task. Rather than learning a monolithic function all at once, the learner can exploit intermediate structure and recover the task node by node. Technically, this leads to efficient SQ-learning guarantees for a broad family of structured problems, together with a constructive procedure for recovering both the underlying DAG and the associated truth tables.

Our results also help separate two questions that are often conflated in discussions of CoT. One question is whether intermediate reasoning traces improve predictive performance; the other is whether those traces faithfully describe the model's internal reasoning. The present work addresses the first question in a clean learning-theoretic setting. By abstracting away from architecture-specific details and optimization dynamics, we isolate the computational value of intermediate variables themselves.

Limitations and Future Directions. There are several promising directions for future work. First, it would be valuable to relax some of the structural assumptions used here, such as bounded fan-in or the lower-bound condition on parent marginals, and to determine how robust the conclusions are beyond the present setting. Second, it would be interesting to extend the analysis to richer intermediate representations, noisy or partially observed CoT signals, and settings in which the reasoning graph itself must be inferred only approximately. Finally, a major open direction is to better understand how these learning-theoretic advantages interact with the actual inductive biases of modern neural architectures and with the practical generation of reasoning traces at inference time.

REFERENCES

Kartik Ahuja and Amin Mansouri. On provable length and compositional generalization. *arXiv preprint arXiv:2402.04875*, 2024.

Avrim Blum, Merrick Furst, Jeffrey Jackson, Michael Kearns, Yishay Mansour, and Steven Rudich. Weakly learning DNF and characterizing statistical query learning using fourier analysis. In *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing*, pages 253–262, 1994.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. *ArXiv preprint*, 2022.

Gilad Deutch, Nadav Magar, Tomer Natan, and Guy Dar. In-context learning and gradient descent revisited. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2024.

Vitaly Feldman. A complete characterization of statistical query learning with applications to evolvability. *Journal of Computer and System Sciences*, 78(5):1444–1459, 2012.

Vitaly Feldman and Varun Kanade. Computational bounds on statistical query learning. In *Proceedings of the 25th Annual Conference on Learning Theory*, 2012.

Nirmit Joshi, Gal Vardi, Adam Block, Surbhi Goel, Zhiyuan Li, Theodor Misiakiewicz, and Nathan Srebro. A theory of learning with autoregressive chain of thought, 2025.

Michael Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 45(6): 983–1006, 1998.

Juno Kim and Taiji Suzuki. Transformers provably solve parity efficiently with chain of thought. In *The Thirteenth International Conference on Learning Representations*, 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*, volume 35, 2022.

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

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

Bingbin Liu, Jordan T. Ash, Surbhi Goel, and Cyril Zhang. Transformers learn shortcuts to automata. In *International Conference on Learning Representations*, 2023.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *ArXiv preprint*, 2021.

Lev Reyzin. Statistical queries and statistical algorithms: Foundations and applications. *ArXiv preprint*, 2020.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, Joao Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, 2023.

Zixuan Wang, Eshaan Nichani, Alberto Bietti, Alex Damian, Daniel Hsu, Jason D Lee, and Denny Wu. Learning compositional functions with transformers from easy-to-hard data. In *Proceedings of the Thirty-Eighth Conference on Learning Theory*, volume 291 of *Proceedings of Machine Learning Research*. PMLR, 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.

Noam Wies, Yoav Levine, and Amnon Shashua. Sub-task decomposition enables learning in sequence to sequence tasks. In *The Eleventh International Conference on Learning Representations*, 2023.

Chenxiao Yang, Zhiyuan Li, and David Wipf. Chain-of-thought provably enables learning the (otherwise) unlearnable. In *The Thirteenth International Conference on Learning Representations*, 2025.

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

A Some Classical Facts

A.1 Orthogonality and Query-Complexity Lower Bounds

Our lower bounds without CoT follow from a direct orthogonality argument underlying classical SQ lower bounds (Blum et al., 1994). The argument applies to arbitrary bounded endpoint-only queries, not merely correlation queries.

Fact 2 (Endpoint-Only SQ Lower Bound). *Let $\mathcal{F} = \{f_1, \dots, f_M\}$ be a family of functions from $\{-1,1\}^d$ to $\{-1,1\}$ that are pairwise orthogonal under a distribution D:*

$$\mathbb{E}_{X\sim D}[f_i(X)f_j(X)] = 0, \text{ for every } i \neq j. \tag{12}$$

Consider an adaptive, possibly randomized learner whose target is an unknown $f_i \in \mathcal{F}$ and whose queries are arbitrary functions $\phi : \{-1,1\}^d \times \{-1,1\} \to [-1,1]$. Suppose that, for every target f_i and every valid $\mathrm{STAT}(\tau)$ oracle, the learner makes at most Q queries and outputs a hypothesis $h : \{-1,1\}^d \to [-1,1]$ satisfying the correlation condition:

$$\mathbb{E}_{X\sim D}[h(X)f_i(X)] \geq \rho \tag{13}$$

with probability at least $2/3$. Then $Q \geq \tau^2(2M/3 - 1/\rho^2)$ queries. In particular, exact identification requires $Q \geq \tau^2\,(2M/3 - 1)$ queries.

Proof. For any query ϕ, define

$$a_\phi(x) := \frac{\phi(x,1) + \phi(x,-1)}{2}, \quad b_\phi(x) := \frac{\phi(x,1) - \phi(x,-1)}{2}.$$

Then $\phi(x,y) = a_\phi(x) + y b_\phi(x)$ and $|b_\phi(x)| \leq 1$. Consequently, when the target is f_i,

$$\mathbb{E}[\phi(X, f_i(X))] = \mathbb{E}[a_\phi(X)] + \langle b_\phi, f_i\rangle_D,$$

where $\langle g, h\rangle_D := \mathbb{E}_{X\sim D}[g(X)h(X)]$. Because the functions $f_1, \dots, f_M$ are orthonormal in $L_2(D)$, Bessel's inequality gives

$$\sum_{i=1}^{M} \langle b_\phi, f_i\rangle_D^2 \leq \|b_\phi\|_{L_2(D)}^2 \leq 1.$$

It follows that

$$H_\phi := \{i \in [M] : |\langle b_\phi, f_i\rangle_D| > \tau\}$$

has cardinality at most $1/\tau^2$.

For each target f_i, consider the oracle that, on query ϕ, returns

$$\mathbb{E}[a_\phi(X)]$$

whenever $i \notin H_\phi$, and returns the exact expectation

$$\mathbb{E}[\phi(X, f_i(X))]$$

otherwise. This is a valid $\mathrm{STAT}(\tau)$ oracle.

Fix the learner's internal randomness and consider the transcript obtained by answering every query ϕ with $\mathbb{E}[a_\phi(X)]$. Let $\phi_1, \dots, \phi_q$, where $q \leq Q$, be the queries along this transcript, and define

$$H := \bigcup_{t=1}^{q} H_{\phi_t}.$$

Since $|H_{\phi_t}| \leq 1/\tau^2$,

$$|H| \leq Q/\tau^2.$$

For every $i \notin H$, the oracle for target f_i produces exactly this transcript. The learner therefore outputs the same hypothesis h for every such target. Another application of Bessel's inequality yields

$$\sum_{i=1}^{M} \langle h, f_i\rangle_D^2 \leq \|h\|_{L_2(D)}^2 \leq 1.$$

Hence h has correlation at least ρ with at most $1/\rho^2$ targets. For the fixed internal randomness, the learner can therefore succeed on at most

$$\frac{Q}{\tau^2} + \frac{1}{\rho^2}$$

of the M target-oracle pairs. Averaging over the learner's randomness, its average success probability over a uniformly random target is at most

$$\frac{Q/\tau^2 + 1/\rho^2}{M}.$$

Since the learner succeeds with probability at least $2/3$ for every target and every valid oracle,

$$\frac{2}{3} \le \frac{Q/\tau^2 + 1/\rho^2}{M}.$$

Rearranging proves the first claim. Exact identification corresponds to $\rho = 1$. □

For a Boolean hypothesis h,

$$\Pr_{X\sim D}(h(X) \neq f_i(X)) = \frac{1 - \mathbb{E}_{X\sim D}[h(X)f_i(X)]}{2}.$$

Thus, error at most $1/2 - \varepsilon$ corresponds to correlation at least 2ε.

For example, let

$$\mathcal{P}_d := \{\chi_A : A \subseteq [d]\}, \quad \chi_A(x) := \prod_{j\in A} x_j,$$

and let μ_d be the uniform distribution on $\{-1, 1\}^d$. For distinct $A, B \subseteq [d]$,

$$\mathbb{E}_{X\sim\mu_d}[\chi_A(X)\chi_B(X)] = 0.$$

Fact 2 with $M = 2^d$ therefore shows that exact identification of an unknown parity from endpoint-only queries with success probability at least $2/3$ requires

$$Q \ge \tau^2 \left(\frac{2^{d+1}}{3} - 1\right).$$

More generally, achieving correlation at least ρ requires

$$Q \ge \tau^2 \left(\frac{2^{d+1}}{3} - \frac{1}{\rho^2}\right).$$

The same argument for the fixed-weight class

$$\mathcal{P}_{d,k} := \{\chi_A : A \subseteq [d],\ |A| = k\}$$

gives

$$Q \ge \tau^2 \left(\frac{2}{3}\binom{d}{k} - \frac{1}{\rho^2}\right).$$

In particular, when $k = \lfloor d/2 \rfloor$ and $\tau, \rho \ge 1/\operatorname{poly}(d)$, the no-CoT query complexity is exponential in d.

A.2 Proof of Fact 1 (SQ-to-Sample Simulation)

For each query ϕ_t, draw a fresh batch of

$$M = \left\lceil \frac{2}{\tau^2} \log \frac{2Q}{\delta} \right\rceil$$

independent observations $W_{t,1}, \ldots, W_{t,M}$ and return

$$\widehat{\phi}_t := \frac{1}{M} \sum_{j=1}^{M} \phi_t(W_{t,j}).$$

Conditional on the preceding transcript, ϕ_t is fixed and the new batch is independent. Hoeffding's inequality gives

$$\Pr\left(\left|\widehat{\phi}_t - \mathbb{E}[\phi_t(W)]\right| > \tau\right) \le 2e^{-M\tau^2/2} \le \frac{\delta}{Q}.$$

A union bound over the at most Q queries proves the claim. □

B Proof of Theorem 1

B.1 Recovery of Parent Set of a Non-Input Node

We start with the following lemma.

Lemma 1. *Under Condition 1, in **Step 1** of the algorithm, a given S passes the TV test $\hat{\Delta}(u,S) > 1-\beta$ for a non-input node u iff $e(u,S)=0$, i.e., iff u is a deterministic boolean function of V_S almost surely.*

Proof. Let $r=|S|$. Define $\hat{d}(a) := \hat{\mu}^+(u,S,a) - \hat{\mu}^-(u,S,a)$ and $d(a) := \mu^+(u,S,a) - \mu^-(u,S,a)$ for any $a \in \{-1,1\}^{|S|}$ and observe that

$$|\hat{d}(a) - d(a)| \le 2\tau, \quad |\hat{\Delta}(u,S) - \Delta(u,S)| \le \sum_a |\hat{d}(a) - d(a)| \le 2^{r+1}\tau \le 2^{s+1}\tau \le \beta/2. \quad (14)$$

Now, if $e(u,S)=0$, then $\Delta(u,S)=1$ from (4), and so

$$\hat{\Delta}(u,S) \ge \Delta(u,S) - \beta/2 = 1 - \beta/2 > 1-\beta,$$

and so S passes the TV test. On the other hand, suppose $e(u,S)>0$. Then, thanks to Condition 1, we have $e(u,S) \ge \beta$. Thus, $\Delta(u,S) \le 1-2\beta$ from (4), and so

$$\hat{\Delta}(u,S) \le \Delta(u,S) + \beta/2 \le 1 - 2\beta + \beta/2 = 1 - 3\beta/2 < 1-\beta,$$

and so S does not pass the TV test. □

B.2 Recovery of Gates

We next show that if S is accepted as a viable parent set for a non-input node u, then $V_u = \hat{g}_{u,S}(V_S)$ almost-surely, i.e

$$\Pr(V_u = \hat{g}_{u,S}(V_S)) = 1, \quad (15)$$

where $\hat{g}_{u,S}$ is the boolean function defined in (7) of **Step 1** of the algorithm. Since S is accepted, it passes the TV test. By Lemma 1, this implies $e(u,S)=0$. Therefore,

$$\min\{\mu^+(u,S,a), \mu^-(u,S,a)\} = 0 \quad \text{for every } a \in \{-1,1\}^{|S|}.$$

Define

$$p_a := \Pr(V_S = a) = \mu^+(u,S,a) + \mu^-(u,S,a), \quad \hat{p}_a := \hat{\mu}^+(u,S,a) + \hat{\mu}^-(u,S,a).$$

Since each estimate has error at most τ,

$$|\hat{p}_a - p_a| \le 2\tau.$$

Because S is accepted, it also passes the empirical coverage test, so

$$\hat{p}_a > \gamma/2 \quad \text{for every } a.$$

It follows that

$$p_a \ge \hat{p}_a - 2\tau > \frac{\gamma}{2} - \frac{\gamma}{8} = \frac{3\gamma}{8},$$

where we used $\tau \le \gamma/16$. Exactly one of $\mu^+(u,S,a)$ and $\mu^-(u,S,a)$ therefore equals p_a, and the other equals zero. Thus,

$$|d(a)| := |\mu^+(u,S,a) - \mu^-(u,S,a)| = p_a > \frac{3\gamma}{8}.$$

On the other hand,

$$|\hat{d}(a) - d(a)| \le 2\tau \le \frac{\gamma}{8} < |d(a)|.$$

Therefore, $\hat{d}(a)$ and $d(a)$ have the same sign. By the definition of $\hat{g}_{u,S}$, this implies

$$\Pr(V_u = \hat{g}_{u,S}(a) \mid V_S = a) = 1.$$

Since this holds for every a, (15) follows.

B.3 Step 2 Terminates in k Rounds

If all intermediate nodes belong to R, then **Step 2** terminates immediately. Otherwise, let u be the earliest intermediate node that does not belong to R, with respect to a topological order on the true DAG G. Every true parent of u is either an input node or an earlier intermediate node that belongs to R. It follows that $\mathrm{par}(u) \subseteq R$.

On the other hand, since u is a deterministic function of its true parents,

$$e(u, \mathrm{par}_G(u)) = 0.$$

Hence, by Lemma 1, $S = \mathrm{par}_G(u)$ passes the TV test (6) and is therefore accepted as viable for u. Moreover, Condition 2 gives

$$\Pr(V_{\mathrm{par}_G(u)} = a) \ge \gamma$$

for every configuration $a \in \{-1, 1\}^{|S|}$. Therefore,

$$\hat{p}_a \ge p_a - 2\tau \ge \gamma - \frac{\gamma}{8} = \frac{7\gamma}{8} > \frac{\gamma}{2}.$$

Thus, $\mathrm{par}_G(u)$ also passes the empirical coverage test (6) and is accepted. Consequently, at least one intermediate node is eligible to be added to R at every unfinished round, increasing the cardinality of R by 1. Since there are k intermediate nodes in the true DAG, **Step 2** stops after exactly k additions.

B.4 The Algorithm Recovers an Equivalent DAG

It remains to show that the learned DAG $\hat{G}$ has the same node values almost-surely as the true DAG G, i.e.,

$$\Pr(\hat{V}_u \ne V_u) = 0 \quad \text{for all non-input nodes } u. \tag{16}$$

To this end, suppose the non-input nodes are added in the order $u_1, \ldots, u_{k+m}$, with the output nodes added after the intermediate nodes. We prove by induction on t that induction on t that

$$\Pr(\hat{V}_{u_t} \ne V_{u_t}) = 0.$$

For the induction base, note that when u_1 is added, all nodes in its selected parent set are input nodes and are therefore correct by construction. More generally, node u_t is added using

$$S_t \subseteq \mathcal{I} \cup \{u_1, \ldots, u_{t-1}\},$$

where $\mathcal{I}$ is the set of input nodes. By the induction hypothesis,

$$\Pr(\hat{V}_v \ne V_v) = 0 \quad \text{for every } v \in S_t.$$

It follows that

$$\Pr(\hat{V}_{S_t} \ne V_{S_t}) \le \sum_{v \in S_t} \Pr(\hat{V}_v \ne V_v) = 0.$$

Thus, $\hat{V}_{S_t} = V_{S_t}$ almost surely. Combining this with (15), we obtain, outside a null event,

$$\hat{V}_{u_t} = \hat{g}_{u_t,S_t}(\hat{V}_{S_t}) = \hat{g}_{u_t,S_t}(V_{S_t}) = V_{u_t}.$$

Therefore, $\Pr(\hat{V}_{u_t} \ne V_{u_t}) = 0$, which completes the induction and establishes (16).

If D has full support, then (16) also implies pointwise equivalence: $\hat{G}(x) = G(x)$ for every input x.

B.5 Total Number of Queries

For every candidate set S of size $r \le s$, the algorithm issues 2^{r+1} queries, one for each $(a, b) \in \{-1, 1\}^r \times \{-1, 1\}$. For every intermediate node, there are at most $\binom{d+k-1}{r}$ choices for S, while for every output node, there are at most $\binom{d+k}{r}$ choices for S. Therefore, the total number of $\mathrm{STAT}(\tau)$ queries Q used in **Step 1** is bounded by

$$Q \le k \sum_{r=0}^{s} 2^{r+1} \binom{d+k-1}{r} + m \sum_{r=0}^{s} 2^{r+1} \binom{d+k}{r} \le (k+m)(s+1)2^{s+1}(d+k)^s,$$

as claimed. If $s = O(1)$, then we further have the polynomial bound $Q \le (s+1)(2N)^{s+1} = \mathrm{poly}(N)$, where $N := d + k + m$ is the total number of nodes in the DAG. □

C Proof of Corollary 1 (Learning Parity with CoT)

We verify Conditions 2 and 1 and then apply Theorem 1.

Let $A = \{i_1, \ldots, i_k\} \subseteq [n]$ be the unknown parity set and use the prefix trace

$$Z_t = \prod_{q=1}^{t} X_{i_q}, \quad t \in [k],$$

with output $Y = Z_k$. Each nontrivial gate has fan-in at most 2. Under the uniform input distribution, Z_{t-1} and X_{i_t} are independent uniform signs, because they are parities of disjoint nonempty sets of input variables. Therefore, every configuration of every true parent set has probability at least $1/4$, and Condition 2 holds with $\gamma = 1/4$. Every input, intermediate, and output node is a parity character. Associate each such node v with its coefficient vector $\alpha_v \in \mathbb{F}_2^n$. Let u be a non-input node and let W be any candidate set with $|W| \le 2$. If $\alpha_u \in \mathrm{span}_{\mathbb{F}_2}\{\alpha_v : v \in W\}$, then V_u is a deterministic function of V_W, and hence $e(u, W) = 0$. Otherwise, α_u is linearly independent of the coefficient vectors in W after redundant vectors are removed. Uniformity of the input then implies that V_u is uniform and independent of V_W, so $e(u, W) = 1/2$. Thus, Condition 1 holds with $\beta = 1/2$. Since $s = 2$, Theorem 1 applies whenever $\tau < \min(\beta/2^{s+1}, \gamma/4) = \min(1/16, 1/16) = \frac{1}{16}$.

Furthermore, the query count upper-bound in Theorem 1 is $Q = O\left((k+1)(n+k)^2\right) = O\left(k(n+k)^2\right) = O(kn^2)$ as claimed. □

D Proof of Corollary 3 (Learning Decision-Tree Traversal with CoT)

Let $p(v)$ denote the tree parent of a non-root vertex v, and let $b_v \in \{0, 1\}$ denote the label of the edge from $p(v)$ to v. The always-active root is folded into the gates of its children. If v has depth one, then $V_v = \mathbf{1}\{X_{q(p(v))} = b_v\}$. If v has depth at least two, then $V_v = V_{p(v)}\mathbf{1}\{X_{q(p(v))} = b_v\}$. Thus every gate has fan-in at most $s = 2$. There are $k = 2^h - 2$ intermediate nodes and $m = 2^h$ output nodes.

Consider a vertex v of depth $t \ge 2$. Since no input coordinate is repeated along a path, $X_{q(p(v))}$ is independent of $V_{p(v)}$, and

$$\Pr(V_{p(v)} = 1) = 2^{-(t-1)}.$$

Consequently, for every $a, b \in \{0, 1\}$,

$$\Pr(V_{p(v)} = 1, X_{q(p(v))} = b) = 2^{-t}, \quad \Pr(V_{p(v)} = 0, X_{q(p(v))} = b) = \frac{1 - 2^{-(t-1)}}{2} \ge 2^{-t}.$$

The depth-one gates have parent-configuration probabilities $1/2$. Hence every configuration of every true parent set has probability at least

$$\gamma = 2^{-h}.$$

It remains to verify the separation condition. Every activation variable depends on at most h input coordinates. Therefore, for any non-input node u and any candidate set S with $|S| \le 2$, the tuple

(V_u, V_S) depends on at most $3h$ independent uniform input bits. Every positive probability of the form $\Pr(V_u = b, V_S = a)$ is consequently at least 2^{-3h}. Using

$$e(u, S) = \sum_a \min\{\Pr(V_u = 0, V_S = a), \Pr(V_u = 1, V_S = a)\},$$

we see that if $e(u, S) > 0$, then at least one summand is positive and hence

$$e(u, S) \geq 2^{-3h}.$$

Thus the separation condition holds with $\beta = 2^{-3h}$.

Theorem 1, with $s = 2$, applies whenever

$$\tau < \min\left\{\frac{\beta}{2^{s+1}}, \frac{\gamma}{4}\right\} = 2^{-(3h+3)}.$$

The choice $\tau = 2^{-(3h+4)}$ therefore suffices. Its query bound becomes

$$O\left((k+m)(d+k)^2\right) = O\left(2^h(d+2^h)^2\right).$$

Finally, the uniform input distribution has full support, so the almost-sure recovery guarantee is pointwise: the learned DAG is correct on every input.

E Rounded Bilinear Matrix Products

Proof. Choose an irreducible polynomial $p(t) \in \mathbb{F}_2[t]$ of degree n, let C be its companion matrix, and define

$$\mathcal{M}_n := \{q(C) : q \in \mathbb{F}_2[t],\ \deg(q) < n\},$$

where the zero polynomial is included. Since the minimal polynomial of C is p, the matrices in $\mathcal{M}_n$ are distinct, and hence $|\mathcal{M}_n| = 2^n$.

Let $M = q(C)$ and $N = r(C)$ be distinct. Then $q + r$ is a nonzero polynomial of degree less than n. Since p is irreducible,

$$\gcd(q + r, p) = 1.$$

Bézout's identity gives polynomials $a, b \in \mathbb{F}_2[t]$ such that

$$a(q + r) + bp = 1.$$

Evaluating at C and using $p(C) = 0$ gives

$$a(C)(M + N) = I.$$

Thus $M + N$ is invertible.

We next describe the CoT learner. For every $(i, j) \in [n]^2$, issue the query

$$\psi_{ij} := (-1)^{Z_{i,j} + Z_{i,j-1}}.$$

By construction,

$$Z_{i,j} + Z_{i,j-1} = M_{ij} Y_j.$$

Since Y_j is uniform on $\mathbb{F}_2$,

$$\mathbb{E}[\psi_{ij}] = \begin{cases} 1, & M_{ij} = 0, \\ 0, & M_{ij} = 1. \end{cases}$$

The learner sets $\widehat{M}_{ij} = 0$ if the oracle response is greater than $1/2$ and sets $\widehat{M}_{ij} = 1$ otherwise. Because $\tau < 1/2$, every valid oracle response lies on the correct side of the threshold. Hence $\widehat{M} = M$ after exactly n^2 queries. Substituting $\widehat{M}$ into the displayed recurrences produces a DAG computing both the trace and f_M exactly.

It remains to prove the endpoint-only lower bound. Write $L := |\mathcal{M}_n| = 2^n$ and define

$$\langle g, h \rangle := \mathbb{E}[g(X, Y)h(X, Y)].$$

For distinct $M, N \in \mathcal{M}_n$,

$$\langle f_M, f_N \rangle = \mathbb{E}\left[(-1)^{X^\top (M+N)Y}\right].$$

Conditioning on Y, the expectation over X is 1 when $(M+N)Y = 0$ and 0 otherwise. Therefore,

$$\langle f_M, f_N \rangle = \Pr((M+N)Y = 0) = 2^{-\operatorname{rank}(M+N)} = 2^{-n}.$$

Thus the Gram matrix $G = (\langle f_M, f_N \rangle)_{M,N \in \mathcal{M}_n}$ is

$$G = (1 - 2^{-n})I + 2^{-n}\mathbf{1}\mathbf{1}^\top.$$

Its largest eigenvalue is

$$\lambda := 1 + (L-1)2^{-n} = 2 - 2^{-n}.$$

Consequently, every function b with $\|b\|_2 \le 1$ satisfies

$$\sum_{M \in \mathcal{M}_n} \langle b, f_M \rangle^2 \le \lambda.$$

Indeed, if $\mathcal{A} : \mathbb{R}^L \to L_2$ is defined by

$$\mathcal{A}c := \sum_{M \in \mathcal{M}_n} c_M f_M,$$

then $\mathcal{A}^*\mathcal{A} = G$, and hence

$$\|\mathcal{A}^* b\|_2^2 \le \|G\|_{\mathrm{op}} \|b\|_2^2 = \lambda \|b\|_2^2.$$

Now consider an arbitrary endpoint-only query

$$\phi : \mathbb{F}_2^n \times \mathbb{F}_2^n \times \{-1, 1\} \to [-1, 1].$$

Define

$$a(x,y) := \frac{\phi(x,y,1) + \phi(x,y,-1)}{2}, \qquad b(x,y) := \frac{\phi(x,y,1) - \phi(x,y,-1)}{2}.$$

Then $\phi(x,y,z) = a(x,y) + z b(x,y)$ and $|b(x,y)| \le 1$. Under target f_M, the query expectation is

$$\mathbb{E}[a(X,Y)] + \langle b, f_M \rangle.$$

Therefore, the target-independent answer $\mathbb{E}[a(X,Y)]$ is a valid tolerance-τ response for every target outside

$$H_\phi := \{M \in \mathcal{M}_n : |\langle b, f_M \rangle| > \tau\}.$$

Since $\|b\|_2 \le 1$,

$$|H_\phi| \le \frac{\lambda}{\tau^2}.$$

For every target M, fix the valid oracle that returns $\mathbb{E}[a(X,Y)]$ whenever $M \notin H_\phi$ and returns the exact target expectation otherwise. Fix the learner's internal randomness and follow the transcript obtained by returning the target-independent answer to every query. If the learner makes queries $\phi_1, \dots, \phi_q$ on this transcript, where $q \le Q$, then

$$\left| \bigcup_{t=1}^{q} H_{\phi_t} \right| \le \frac{Q\lambda}{\tau^2}.$$

Every target outside this union produces the same transcript and hence the same output. Since the functions f_M are distinct, this output can identify at most one such target. Thus, for the fixed internal randomness, the learner succeeds on at most

$$\frac{Q\lambda}{\tau^2} + 1$$

targets. Averaging over the learner's randomness and a uniformly random target gives

$$\frac{2}{3} \le \frac{Q\lambda/\tau^2 + 1}{L}.$$

Substituting $L = 2^n$ and $\lambda = 2 - 2^{-n}$ and rearranging yields

$$Q \ge \frac{\tau^2}{2 - 2^{-n}} \left(\frac{2^{n+1}}{3} - 1 \right).$$

□

F TECHNICAL LEMMAS

Lemma 2 (Separation for the ripple-carry trace)**.** *Let $(A_i, B_i)_{i=0}^{n-1}$ be mutually independent uniform bits, and define*

$$C_0 = 0, \quad T_i = A_i \oplus B_i, \quad S_i = T_i \oplus C_i, \quad C_{i+1} = \mathrm{Maj}(A_i, B_i, C_i).$$

The intermediate nodes are $T_0, \dots, T_{n-1}, C_1, \dots, C_{n-1}$, and the output nodes are $S_0, \dots, S_{n-1}, C_n$. For every non-input node u and every admissible candidate set W of at most three input or intermediate nodes,

$$e(u, W) = 0 \quad \textit{or} \quad e(u, W) \geq 1/8.$$

Proof. Put $P_i := A_i \oplus B_i$. Then (A_i, P_i) is uniform on $\{0,1\}^2$, independently across i, and

$$C_{i+1} = \begin{cases} A_i, & P_i = 0, \\ C_i, & P_i = 1. \end{cases}$$

Consequently, the carry process has transition matrix

$$K = \frac{1}{4}\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix},$$

and, for every integer $\ell \geq 1$,

$$K^\ell = \frac{1}{2}\begin{pmatrix} 1 + 2^{-\ell} & 1 - 2^{-\ell} \\ 1 - 2^{-\ell} & 1 + 2^{-\ell} \end{pmatrix}.$$

For a conjunction E of assigned node values at bit position i, define

$$L_E(c, c') := \frac{1}{4} \sum_{a,b \in \{0,1\}} \mathbf{1}\{E(a, b, c)\}\mathbf{1}\{c' = \mathrm{Maj}(a, b, c)\},$$

where the values of the local trace variables inside E are

$$T_i = a \oplus b, \quad S_i = a \oplus b \oplus c.$$

Fix a target u, a candidate set W with $|W| \leq 3$, and an assignment to (V_u, V_W). Group the resulting constraints by bit position, and let $i_1 < \cdots < i_q$ be the marked positions. Since (V_u, V_W) contains at most four variables, $q \leq 4$. The corresponding joint probability has the form

$$e_0^\top K^{i_1} L_{E_1} K^{i_2 - i_1 - 1} L_{E_2} \cdots K^{i_q - i_{q-1} - 1} L_{E_q} \mathbf{1},$$

with the appropriate terminal selector when C_n is constrained, where $e_0 = (1, 0)^\top$. If two marked positions are adjacent, the corresponding intervening factor is $K^0 = I$.

For every positive exponent ℓ, write

$$K^\ell = K(\rho_\ell), \quad K(\rho) := \frac{1}{2}\begin{pmatrix} 1 + \rho & 1 - \rho \\ 1 - \rho & 1 + \rho \end{pmatrix}, \quad \rho_\ell := 2^{-\ell} \in (0, 1/2].$$

Every joint probability $\Pr(V_u = b, V_W = a)$ is affine in each gap parameter separately. Therefore,

$$e(u, W) = \sum_a \min\{\Pr(V_u = 0, V_W = a), \Pr(V_u = 1, V_W = a)\}$$

is separately concave in the gap parameters. Iterating one-dimensional concavity gives

$$e(u, W) \geq \min_{\boldsymbol{\varepsilon} \in \{0, 1/2\}^q} e_{\boldsymbol{\varepsilon}}(u, W).$$

Every matrix $K(\rho)$ is entrywise positive for $\rho \in [0, 1/2]$. Since every factor in the matrix products is nonnegative, replacing any positive-gap parameter by 0 or $1/2$ does not change whether a given joint probability is zero. Hence, if $e(u, W) > 0$, then every endpoint error on the right-hand side is also positive.

At an endpoint, each positive-gap factor is either

$$K = \frac{1}{4}\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} \quad \text{or} \quad J = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix},$$

while each zero-gap factor remains I. Thus only finitely many cases remain. They are obtained by choosing the target type T_i, C_i, or S_i; choosing at most three distinct candidate nodes of types A_j, B_j, T_j, or C_j; grouping these nodes into at most four ordered marked positions; choosing I, K, or J between successive marked positions; and enumerating their Boolean assignments. Substitution into the displayed definition of L_E gives the following least strictly positive endpoint errors:

	$\lvert W\rvert = 0$	$\lvert W\rvert = 1$	$\lvert W\rvert = 2$	$\lvert W\rvert = 3$
$u = T_i$	$1/2$	$1/4$	$1/4$	$1/4$
$u = C_i$	$1/4$	$1/4$	$1/8$	$1/8$
$u = S_i$	$1/2$	$1/4$	$1/8$	$1/8$.

The C_i row includes the final carry C_n and the boundary cases. Therefore, every positive endpoint error is at least $1/8$. It follows that $e(u, W) > 0 \implies e(u, W) \geq 1/8$ as claimed. □