REASONING MODELS ARE ACCURATE BUT UNSOUND ON IDENTIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

A reasoning model asked whether a causal effect is recoverable from observational data can fail in two ways: it refuses an identifiable query or answers a non-identifiable one. The latter is more consequential, as no observational data can validate the claimed formula. Measuring this failure requires queries that are provably non-identifiable, which prior evaluations lack, and grading that accepts correct formulas in any equivalent form, which string matching cannot provide.

We build CERTID, a formal identification pipeline that addresses both limitations. CERTID uses the sound and complete causal identification algorithm ID to certify whether an effect is identifiable from a given graph and query, and verifies returned formulas against structural causal models whose interventional distributions are known exactly. CERTID further develops theoretical results to mitigate structural leakage, repair non-identifiable queries, and establish grading guarantees.

We evaluate three frontier reasoning models (Gemini Flash, Gemini Pro, and GPT-5.5) on 1,200 certified instances spanning 4 to 50 vertices. Accuracy proves a poor proxy for soundness: on identical instances, the false-claim rate on non-identifiable queries varies by seventeen-fold across models. We also find that models decide identifiability with 97–100% accuracy on graphs generated after the strongest model's training snapshot. Instances, the certification procedure, the verifier, and per-instance records are available at `https://anonymous.4open.science/r/certid-D718`.

1 INTRODUCTION

Evaluating causal reasoning requires fixing what counts as a correct and sound answer, and current benchmarks do so in one of two ways. **Value-computing** benchmarks (Jin et al., 2023; Xu & Fu, 2026; Liu et al., 2024) generate questions from a fully specified structural causal model (SCM) and take the computed quantity as ground truth. Every question such a benchmark poses is answerable by construction. **Annotation-based** benchmarks (Wang, 2024; Kirichenko et al., 2026; Lee et al., 2026) label correctness, including whether a question can be answered at all, by human judgment. Both families then score a response by matching it against a reference string. Both approaches are defensible, yet their notions of ground truth leave a specific failure mode unmeasured.

Consider a practitioner asking whether a treatment affects an outcome when the two share an unmeasured common cause (Figure 1(a)). In this graph, the interventional quantity $P(y \mid \mathrm{do}(x))$ is not identifiable: no functional of the observational distribution equals it, regardless of sample size or estimator (Shpitser & Pearl, 2006b). Bounds may remain recoverable, but the quantity itself cannot be recovered from observational data alone. A model that answers anyway may return an incorrect formula whose error no amount of observational data can expose. This is precisely the failure that matters in deployment, yet existing benchmark families do not measure it. Value-computing benchmarks contain only identifiable queries, while annotation-based benchmarks rely on human judgments of identifiability, on which annotators may disagree.

The central question we address is **whether a causal reasoning benchmark can compute its own ground truth, so that neither the label nor the grade depends on human judgment** (see Table 1 for a comparison with prior work). Formal identification answers this question (Figure 1(b)). In particular, the ID dentification algorithm (Shpitser & Pearl, 2006b) can answer causal queries; we use it instead to *label* them, providing formal ground truth for evaluating systems that answer such

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.

Table 1: How ground truth and grading are established in causal reasoning benchmarks. In prior methods, every query is answerable when groundtruth is provided or unanswerable determined by human judgment. CERTID is the only benchmark without ground truth or human judgment.

Benchmark	Ground truth	Unanswerable instances	Grading
CLADDER (Jin et al., 2023)	oracle-computed	—	string
NOISYCAUSAL (Xu & Fu, 2026)	SCM-computed	—	string
QRDATA (Liu et al., 2024)	data-computed	—	string
CAUSALBENCH (Wang, 2024)	annotated	—	string
ECONCAUSAL (Lee et al., 2026)	curated	underspecified	string
ABSTENTIONBENCH (Kirichenko et al., 2026)	annotated	annotated	string
CRB (Sawarni et al., 2026)	curated	—	specification
CERTID (ours)	**decided by ID**	**decided by ID**	**numerical**

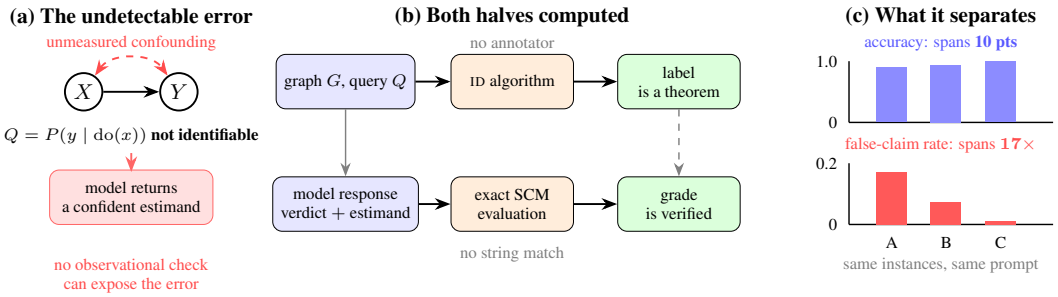


Figure 1: **(a)** an error current evaluations cannot catch, **(b)** the two computations that let us catch it, and **(c)** what the resulting measurement separates.

queries. This removes human judgment from ground-truth construction and enables instances to be generated at arbitrary scale after a model has been trained.

Given an acyclic directed mixed graph G over observed variables $\mathbf{V}$ and a query $Q = P(\mathbf{y} \mid \mathrm{do}(\mathbf{x}))$, ID decides in time polynomial in $|\mathbf{V}|$ whether Q is identifiable from the observational distribution. The algorithm is sound and complete (Shpitser & Pearl, 2006b), *so its verdict is a theorem about the query, not an opinion about it*. Grading an identified estimand can likewise be reduced to computation. An estimand is a functional of the observational distribution, so we can instantiate a model consistent with the graph, evaluate the returned estimand, and compare it with the model's true interventional effect. *Agreement is a calculation, not a judgment.*

Having this decision procedure, however, is not enough; two difficulties remain. First, a certified label does not ensure that a task tests what it was built to test. One of our tasks asks which confounding in a graph is real, and the variable names should carry that answer. A natural construction instead leaves structural signatures: a classifier that reads graph structure and ignores the names separates real from spurious confounding at $\mathrm{AUC} = 0.825$. We diagnose this structural leakage and remove most of it in Section 3.2. Second, exact grading requires enumerating interventional distributions with latent variables materialized, which is costly and unforgiving. Applying our verifier to a widely used causal inference library revealed two graphs for which the library returns numerically incorrect estimands even though its identifiability verdicts remain sound (Section 4.3).

We introduce CERTID, a novel framework for formal causal identification. Every label follows from a complete decision procedure, and every response is graded by numerical evaluation. We further develop a leakage diagnosis and remedy, so that tasks requiring domain knowledge cannot be solved from graph structure alone. The task is to determine, given a graph and a causal query, whether the query is identifiable from observational data and, if so, to return an estimand. The input is an acyclic directed mixed graph, represented by a vertex list with directed and bi-directed edges, where each bi-directed edge represents unmeasured confounding, together with a target $P(\mathbf{y} \mid \mathrm{do}(\mathbf{x}))$. The output is an identifiability verdict and, when affirmative, an estimand. Both are scored against formally certified ground truth. The same machinery applies unchanged to published causal models, e.g., on

a causal diagram from Robins et al. (2000) (Figure 5 in Section H.4), CERTID numerically confirms the authors' estimand and shows that their identifying assumption is sufficient but not minimal.

We evaluate three frontier reasoning models (Gemini Flash, Gemini Pro, and GPT-5.5) from two vendors on 1,200 instances. Three key findings emerge. First, identifiability is decided at 97–100% on graphs that postdate the training snapshot of the strongest model tested, ruling out contamination from those graphs. Second, correct verdicts come with no justification by default, yet every model provides one when prompted, showing that observed inspectability depends on how the model is queried. Third, accuracy is a poor proxy for soundness. The third finding has direct consequences for deployment, as Figure 1(c) illustrates. On identical instances, the rate at which models assert answers to provably unanswerable queries varies seventeen-fold across the three frontier models, while their accuracy varies by only ten points. Accuracy alone would therefore make the three models appear similarly capable while concealing a substantial difference in how often they answer when no answer is identifiable.

2 BACKGROUND AND PROBLEM STATEMENT

Random variables are uppercase (X, Y, V), realizations lowercase (x, y, v), and sets boldface $(\mathbf{X}, \mathbf{Y}, \mathbf{V})$. We write G for an acyclic directed mixed graph (ADMG) over observed variables $\mathbf{V}$, in which a directed edge $V_i \to V_j$ encodes direct causation, and a bi-directed edge $V_i \leftrightarrow V_j$ encodes an unobserved common cause. We write $B(G)$ for the bi-directed edges of G and $\mathrm{An}(\cdot)$ for ancestry. An ADMG can be obtained from a DAG with latent variables by *latent projection* (Verma & Pearl, 2022), which removes the latent variables and encodes their effects on the observed ones through directed and bi-directed edges. $\mathcal{M}(G)$ is the set of structural causal models (SCMs) whose structural dependencies are contained in G (Pearl, 2009), and $P(\mathbf{V})$ is the observational distribution. A *query* is an interventional distribution $Q = P_{\mathbf{x}}(\mathbf{y}) = P(\mathbf{y} \mid \mathrm{do}(\mathbf{x}))$ for disjoint $\mathbf{X}, \mathbf{Y} \subseteq \mathbf{V}$: the distribution of $\mathbf{Y}$ when $\mathbf{X}$ is set by intervention. Complete notation appears in Appendix A.1.

Definition 1 (Identifiability)**.** Q is *identifiable* in G if every pair $\mathcal{M}_1, \mathcal{M}_2 \in \mathcal{M}(G)$ inducing the same positive $P(\mathbf{V})$ agrees on Q; equivalently, if Q is a functional of $P(\mathbf{V})$ invariant across $\mathcal{M}(G)$.

Identifiability is a property of (G, Q) alone, determined before any data are observed. When Q is not identifiable, no estimator can recover it from observational data alone, regardless of sample size.

The three rules of do-calculus, i.e., inserting/deleting observations, exchanging actions with observations, and inserting/deleting actions, are sound (Pearl, 1995) and complete (Huang & Valtorta, 2006) for interventional identification, but do not themselves provide a decision procedure for whether a reducing sequence exists. The ID algorithm provides such a procedure (Shpitser & Pearl, 2006b).

Definition 2 (ID algorithm (Shpitser & Pearl, 2006b))**.** Given (G, Q), ID runs in time polynomial in $|\mathbf{V}|$ and returns either a closed-form estimand ϕ in terms of $P(\mathbf{V})$ or FAIL with a graphical witness of non-identifiability. This witness is a *hedge*, whose presence exactly characterizes non-identifiability.

We use hedges only through ID; Appendix A.2 provides the definitions and criterion.

Evaluating causal identification requires a correct label for each instance, a correct grade for each returned estimand, and instances that test the intended reasoning. Definitions 3 and 4 provide the first two, while Definition 5 states the evaluation problem. Sec. 3 constructs and validates the instances.

Definition 3 (Instance and certified label)**.** An *instance* is a pair (G, Q). Its *certified label* is $\ell = \mathrm{ID}(G, Q) \in \{\phi, \mathrm{FAIL}\}$. Because ID is sound and complete, ℓ is determined by (G, Q) alone and requires no human judgment. We call the triple (G, Q, ℓ) a *certified instance*.

String matching rejects correct estimands written in a different form and accepts wrong ones that look similar, so we grade estimands numerically.

Definition 4 (Estimand grade)**.** Let $\hat{\phi}$ be an estimand returned by a model for an identifiable instance, and τ be a numerical tolerance. Draw $\mathcal{M}_1, \ldots, \mathcal{M}_k$ from $\mathcal{M}(G)$ with all latent variables materialized, so that both $P_{\mathcal{M}_j}(\mathbf{V})$ and $Q_{\mathcal{M}_j}$ can be computed exactly. The estimand is *refuted* if $|\hat{\phi}(P_{\mathcal{M}_j}(\mathbf{V})) - Q_{\mathcal{M}_j}| > \tau$ for some j, and *verified over k draws* otherwise.

Definition 5 (Certified identification problem)**.** A system under evaluation maps a serialization of (G, Q) to a response $r = (v, \hat{\phi})$, where $v \in \{\mathrm{ID}, \mathrm{NID}, \perp\}$ is a verdict and $\hat{\phi}$ is present only when

$v = \text{ID}$. Scoring compares the verdict v with the certified label ℓ from Definition 3. If $\ell = \text{FAIL}$, the response is a *correct refusal* when $v = \text{NID}$ and a *false claim* when $v = \text{ID}$. If $\ell = \phi$, the response is a *correct claim* when $v = \text{ID}$, with $\hat{\phi}$ graded according to Definition 4, and a *miss* when $v = \text{NID}$. Responses with $v = \perp$, for which no verdict can be recovered, are *unscored* and reported separately.

The four cases—correct claim, correct refusal, miss, and false claim—are not equally costly. A miss withholds an identifiable answer that the user can obtain, e.g., by running ID. A false claim supplies an answer that does not exist, and no observational data can expose it. Grading is also asymmetric: only a correct claim carries an estimand to verify, while a correct refusal is complete as stated.

3 CERTID: THEORETICAL FOUNDATION

Section 2 defined the problem, its certified labels and estimand grade. This section forms the theoretical foundation. Section 3.1 constructs instances that test the intended reasoning. Section 3.2 checks that the construction does not reveal the answer through graph structure alone, the first difficulty named in Section 1. Section 3.3 adds the repair task on non-identifiable instances. Section 3.4 proves what the estimand grade guarantees, which addresses the second difficulty, exact grading.

3.1 INSTANCE FAMILIES

We use two graph families. In the random family, variable names are meaningless symbols, so the system must reason from graph structure alone. In the published family, graphs and variable names come from published causal diagrams, allowing the system to also draw on domain knowledge.

Definition 6 (Random family)**.** Sample a topological order on $\mathbf{V}$, add directed edges independently with probability p_{dir} and bi-directed edges with probability p_{bi}, and draw $\mathbf{X}, \mathbf{Y}$ with $\mathbf{X} \subseteq \text{An}(\mathbf{Y})$. Variable names are neutral symbols.

Definition 7 (Published family)**.** Take a published DAG D with interpretable variable names and select a latent set L whose parents, if any, also belong to L. Project D onto $\mathbf{V} = V(D) \setminus L$, adding a bi-directed edge between each pair of observed variables that share a parent in L. Because L is root-closed, the projection introduces only bi-directed edges and no additional directed structure.

The published family supports a second task. Alongside the real bi-directed edges induced by projection, we add spurious edges not implied by the published diagram, yielding *semi-synthetic* instances. The *edge-classification* task asks the system to distinguish real from spurious bi-directed edges, using variable names as the intended evidence. Appendix G.6 evaluates small open-weights language models on this task.

Identifiable and non-identifiable instances are balanced by construction, and the natural rate before balancing is reported separately, so class balance is not mistaken for a population property. Because ID is polynomial, instances are generated at any size and density with no annotator in the loop.

Oracle agreement. Every certified label (Definition 3) is computed by two independently developed implementations of ID. Across $\sim 14.9M$ backend calls, the two produced 0 disagreements. Both also reproduce the published verdicts on a hand-encoded set of 24 canonical instances (Appendix F).

3.2 STRUCTURAL LEAKAGE

Edge classification should require the variable names. If graph structure alone reveals which edges are real, a system can answer correctly without domain knowledge. We hence measure such leakage.

Definition 8 (Structural leakage)**.** The *leakage* of a construction is the AUC of a classifier that distinguishes real from spurious bi-directed edges using graph structure alone. Leakage near 0.5 indicates chance-level separability.

Leakage is high when real and spurious edges are generated differently. By Definition 7, a latent induces a bi-directed edge between every pair of its observed children, so real edges form cliques. We therefore generate spurious edges as cliques of invented latents, structurally matched to real ones. The remaining leakage is the baseline a system must exceed to demonstrate use of variable names (Section 4.1; Appendix E). More generally, generating the two edge types differently risks.

3.3 THE REPAIR TASK

Each bi-directed edge encodes an unobserved common cause (Section 2). Assuming some causes are absent corresponds to deleting their edges, which may make a non-identifiable query identifiable. The *repair task* asks for a minimal set of such deletions; every answer could be checked.

Definition 9 (Augmentation and restoring set)**.** An *augmentation* is a set $A \subseteq B(G)$ of bi-directed edges, and $G \ominus A$ is G with those edges deleted, asserting that the corresponding pairs share no unobserved common cause. Define $\mathcal{A}(G, Q) = \{A : Q \text{ is identifiable in } G \ominus A\}$, $\mathcal{F}(G, Q) = \min_{\subseteq} \mathcal{A}(G, Q)$, where $\mathcal{F}(G, Q)$ contains the inclusion-minimal *restoring sets*.

Proposition 1 (Monotonicity)**.** *Deleting any set of edges preserves identifiability, so $\mathcal{A}(G, Q)$ is upward closed; and $B(G) \in \mathcal{A}(G, Q)$, since $G \ominus B(G)$ is Markovian. Hence $\mathcal{F}(G, Q)$ is non-empty and characterizes $\mathcal{A}(G, Q)$.*

Lemma 1 (Greedy minimization)**.** *Let $A \in \mathcal{A}(G, Q)$. Deleting one element at a time, whenever the result remains in $\mathcal{A}$, yields an element of $\mathcal{F}(G, Q)$ in $|A|$ calls to* ID*, for any deletion order.*

Lemma 1 makes every repair answer checkable: validity of a proposed A costs one call to ID, and minimality costs $|A|$ more. Augmentations are restricted to bi-directed edges for a semantic reason. Proposition 1 covers directed deletions, but deleting $X \to Y$ when that edge is the only directed path forces $P(y \mid \mathrm{do}(x)) = P(y)$, which restores identifiability by assuming the answer. For a subgraph F of G, let $F \setminus A$ denote deletion of the edges in A. An augmentation A *destroys* a hedge (F, F') of G if $(F \setminus A, F' \setminus A)$ is not a hedge in $G \ominus A$.

Theorem 1 (Hedge destruction)**.** *Let $\mathcal{H}(G, Q)$ be the hedges (Appendix A.2) for $P_{\mathbf{x}'}(\mathbf{y}')$ in G over all $\mathbf{X}' \subseteq \mathbf{X}$, $\mathbf{Y}' \subseteq \mathbf{Y}$. Then $A \in \mathcal{A}(G, Q)$ if and only if A destroys every element of $\mathcal{H}(G, Q)$. Further, A destroys (F, F') if and only if deleting A disconnects the bi-directed edges of F or of F'.*

Theorem 1 casts repair as an edge-cut problem. The closest prior result concerns interventions: Akbari et al. (2022) show that an intervention set restores identifiability exactly when it hits every hedge, and that the resulting design problem is NP-hard. Deleting edges instead of intervening on vertices changes the covering objects from vertices to edge cuts, so Theorem 1 does not follow their result. When every hedge has a tree-structured bi-directed component, however, disconnection requires deleting a single edge, reducing restoration to minimum hitting set. Morevoer, membership in $\mathcal{A}(G, Q)$ is a connectivity question once the hedges are known, decidable in $O(|\mathbf{V}| + |B(G)|)$ per hedge and without a call to ID. Appendix C.1 gives further results on the frontier size and the cost of deciding membership in $\mathcal{A}(G, Q)$.

3.4 WHAT THE GRADING GUARANTEES

Definition 4 refutes an estimand on a single numerical disagreement. A disagreement is a counterexample, so every refutation is correct. Verification rests on agreement over k draws, and Theorem 2 shows that this agreement is strong evidence.

Theorem 2 (Soundness and almost-sure refutation)**.** *Fix G and an identifiable Q, and let $\hat{\phi}$ be a returned estimand. Parametrize $\mathcal{M}(G)$ by the conditional probability tables θ of a discrete SCM, so that $\hat{\phi}(P_\theta(\mathbf{V}))$ and Q_θ are rational functions of θ. Draw θ from any distribution absolutely continuous on the parameter region. With $\tau = 0$ and exact arithmetic:* (i) *if the procedure refutes $\hat{\phi}$, then $\hat{\phi}$ does not denote Q; and* (ii) *if $\hat{\phi}$ does not agree with Q on this family, then a single draw refutes it with probability one.*

Part (i) is immediate. For part (ii), the difference of the two rational functions is not identically zero, so its zero set has measure zero, following the argument underlying randomized polynomial identity testing (Schwartz, 1980; Zippel, 1979). Section 4.3 gives two restrictions under which our implementation compares only a slice of the target. Appendix C.2 examines how much of Theorem 2 survives finite precision and the single intervention value used in our implementation. Proofs are deferred to Appendix D.

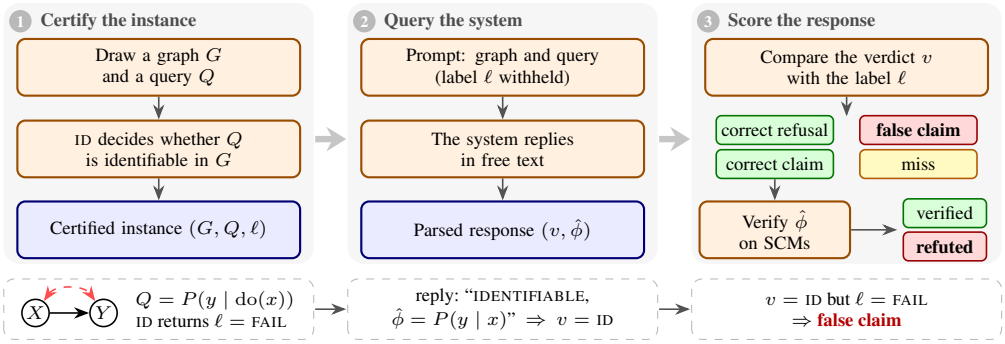


Figure 2: How CERTID's one instance moves through the evaluation. **(1)** An instance is drawn from a family defined in Section 3.1 and certified by ID (Definition 3). **(2)** The system sees the graph and query but not the label, and its reply is parsed into a response (Section 4.2). **(3)** The verdict is compared with the label (Definition 5); for a correct claim, the estimand is also verified on sampled SCMs (Section 4.3). The dashed row follows the confounded graph of Figure 1(a): the system returns an estimand for a non-identifiable query, a false claim that no observational data can expose.

4 CERTID: EVALUATION SETUP

Sections 2 and 3 defined what we measure and why it can be trusted. This section describes the concrete run, in the three stages of Figure 2. Section 4.1 generates and labels the instances, Section 4.2 poses them to the systems and parses the replies, and Sections 4.4 and 4.3 score the responses.

4.1 INSTANCES

We instantiate the two families of Section 3.1. Every instance (Definition 3) pairs a graph G with a query $Q = P(y \mid \mathrm{do}(x))$, where x and y are single vertices and x is a strict ancestor of y. Random-family graphs (Definition 6) fix $p_{\mathrm{dir}} = 0.35$ and sweep p_{bi} over $\{0.10, 0.20, 0.30, 0.40\}$ and $|\mathbf{V}|$ over $\{5, 7, 9, 11, 13\}$, with 24 instances in each of the 20 cells. Published-family graphs (Definition 7) project out latent sets from five published networks—ASIA, CHILD, INSURANCE, ALARM, and HEPAR2 (more details are in Appendix G.1)—and restrict the resulting graph to $\mathrm{An}(y)$. Each network contributes 24 instances with 4–20 vertices.

Both families are balanced by rejection: ID labels each draw, and a draw is discarded once its label's quota in the cell is full. The *main pool* contains 600 instances: 480 random (240 identifiable) and 120 published (60 identifiable). The *scale grid* contains 600 additional random instances, 100 at each $|\mathbf{V}| \in \{10, 15, 20, 30, 40, 50\}$, balanced within each size and containing at least one bi-directed edge. Edge probabilities decrease with $|\mathbf{V}|$ to maintain densities of about 1.55 directed and 1.0 bi-directed edges per vertex. The outcome y is sampled among vertices with the largest ancestor sets, ensuring that the relevant subgraph grows with $|\mathbf{V}|$. The edge-classification task uses a separate set of 200 semi-synthetic instances (Appendix E). Their leakage is AUC $= 0.652$, against 0.825 when spurious edges are placed at random. **The verdict task runs on both pools, the repair task on their non-identifiable instances, and edge classification on the 200 semi-synthetic instances.**

4.2 QUERYING THE SYSTEMS

Each instance is sent to a system as a prompt, and the system's reply is parsed into a response (Definition 5). We query three frontier models from two vendors: `gemini-3.7-flash` and `gemini-3.1-pro-preview`, floating aliases that cannot be re-run against identical weights, and `gpt-5.5-2026-04-23`, a dated snapshot whose served identifier we log on every call. We call them FLASH, PRO and GPT-5.5. Three open-weights models from 0.5B to 3B parameters attempt only edge classification (Appendix G.6). Appendix F.5 lists all identifiers.

The prompt lists the vertices, the directed edges and the bi-directed edges, then states the query and the task. The template is shared across vendors: on the scale grid the prompts are identical in bytes, and on the main pool only the layout of the graph block differs between vendors. The repair task also has an *enumeration* variant, which asks for every minimal set and is scored as recall against $\mathcal{F}(G, Q)$ where exhaustive enumeration is feasible. In the *oracle condition*, the system may call ID as a tool up to 15 times, which separates the ability to search from the ability to check.

The parser reads the verdict by three routes, in order: an explicit verdict line; an estimand block, treated as a claim; or a keyword scan. The second route yields only claims, making the routes asymmetric across labels. Because internal reasoning shares the reply token budget, we use large budgets: 16,384 tokens on the main pool and 32,768 elsewhere. Truncation can bias surviving replies toward easier instances, so we report it per cell: FLASH and GPT-5.5 have none, while PRO has thirteen truncated replies on the main pool and three on the scale grid. Appendix G gives the budgets, serializations, route counts, and results excluding truncated replies.

4.3 ESTIMAND VERIFICATION

Definition 4 is realized as follows. A returned estimand is parsed into a restricted arithmetic form over conditional probabilities and evaluated against exact interventional distributions from randomly drawn discrete SCMs, with latent variables materialized and the joint distribution enumerated.

Example. Consider the front-door graph of Pearl (1995), $X \to M \to Y$ with $X \leftrightarrow Y$, and the query $P(y \mid \mathrm{do}(x))$. Suppose a system returns $\hat{\phi} = \sum_m P(m \mid x) \sum_{x'} P(y \mid x', m)\, P(x')$. The verifier replaces $X \leftrightarrow Y$ with an explicit binary latent U, draws every conditional probability table at random, and enumerates all 16 configurations of (U, X, M, Y). It evaluates $\hat{\phi}$ on the observational margin $P(X, M, Y)$, then computes the true $P(y \mid \mathrm{do}(x))$ by intervening on X in the same SCM. The two agree to $\sim 10^{-16}$, so $\hat{\phi}$ is verified. In contrast, the naive answer $P(y \mid x)$ disagrees: it absorbs the confounding carried by U, and therefore refuted.

We draw $k = 2$ SCMs per instance and refute when $|\hat{\phi}(P_{\mathcal{M}}(\mathbf{V})) - Q_{\mathcal{M}}| > \tau$, with $\tau = 10^{-9}$. In exact arithmetic one draw suffices (reflected in Theorem 2); the second guards against a floating-point coincidence on the first (see more details in Appendix C.2).

We calibrate the verifier in both directions: back-door and front-door estimands agree with ground truth at $\sim 10^{-16}$, and a deliberately naive estimand is refuted at 3.8×10^{-2}. For each library defect of Appendix F.4, the largest deviation over the grader's draws is at least 7.5×10^{-4}.

Exact enumeration is exponential in the number of variables, counting one binary latent per bi-directed edge. We check an estimand only when the graph has at most 9 vertices, the joint has at most 2^{22} configurations, and no conditional probability table exceeds 2^{18} rows.

Restriction. The comparison uses a single intervention value, so an estimand that agrees with the target at $\mathrm{do}(x = 1)$ but differs elsewhere is not refuted. Conditioning events with zero probability yield undefined entries, which are skipped rather than failed. Theorem 2 applies to the quantity actually compared; extending the comparison across intervention values would strengthen the verifier.

What this catches. On five graphs, a published causal inference library returns numerically incorrect expressions despite correct identifiability verdicts. Two of the graphs come from the literature, and the formulas published for them agree with ground truth to $\sim 10^{-16}$. This illustrates the limitation of string matching noted before Definition 4: the library's expression, the published formula, and other correct renderings differ in form (Appendices F.4 and H.3), whereas numerical verification distinguishes them by what they compute.

4.4 METRICS

We report four metrics over the scored responses defined in Definition 5. For a pool, let $\mathcal{S}$ denote its scored instances, $\mathcal{I} \subseteq \mathcal{S}$ the identifiable ones, and $\mathcal{N} = \mathcal{S} \setminus \mathcal{I}$ the non-identifiable ones. **Accuracy:** the fraction of $\mathcal{S}$ answered with a correct claim or correct refusal. **Identifiable rate:** the fraction $|\mathcal{I}|/|\mathcal{S}|$ of scored instances that are identifiable, equivalently the accuracy obtained by calling every query identifiable. **Answer rate:** the fraction of $\mathcal{S}$ answered with verdict ID. **False-claim rate:** the fraction of $\mathcal{N}$ answered with a false claim.

The identifiable rate and answer rate help interpret accuracy. A system that always answers ID achieves an accuracy equal to the identifiable rate; a system that successfully distinguishes identifiable from non-identifiable queries achieves accuracy above this baseline, while its answer rate approaches the identifiable rate. The false-claim rate isolates the consequential error in Definition 5, which accuracy conflates with misses.

5 CERTID: LLM RESULTS

5.1 FRONTIER MODELS DECIDE IDENTIFIABILITY

From Table 2, all three frontier models exceed the identifiable rate, with answer rates close to it, indicating genuine discrimination. PRO's 11 unscored instances comprise 7 main-pool truncations, 1 formatting failure, and 3 transport errors. The truncations reflect the bias noted in Section 4.2: each reaches the budget ceiling after roughly 15,000 tokens of internal reasoning, so the dropped instances are those the model worked longest on. PRO's main-pool accuracy is therefore conditional on the surviving sample.

Table 2: Verdict performance on the main pool: 600 instances, balanced between identifiable and non-identifiable queries. Random-family graphs sweep $|\mathbf{V}| \in \{5, 7, 9, 11, 13\}$.

Model	Scored	Acc.	Id. rate	Ans. rate	False-claim rate
FLASH	600/600	0.983	0.500	0.513	0.030
PRO	589/600	0.978	0.506	0.511	0.028
GPT-5.5	600/600	**0.998**	0.500	0.498	**0.000**

	Accuracy		False-claim	
Model	Rand. fam.	Pub. fam.	Rand. fam.	Pub. fam.
FLASH	0.981	0.992	0.033	0.017
PRO	0.972	**1.000**	0.035	**0.000**
GPT-5.5	**1.000**	0.992	**0.000**	**0.000**

Accuracy on the random family is of primary interest because verdict must follow from graph structure alone (Section 3.1). For GPT-5.5, the timeline rules out contamination: its snapshot dates to April 23, 2026, nearly 4 months before we generated the random instances on August 17, 2026, yet it answers all of them correctly.

Two observations qualify the reading. Performance differs between the families, but the published instances are naturally easier: 97.8%(2,716/2,777) are identifiable before balancing vs 59.4%(585/983) for the random family (Section 3.1). Thus the gap is confounded by difficulty. Gaps are small, and their sign differs between vendors. Gemini models are more accurate on the published instances, while GPT-5.5 is exact on the random instances and has a single error on published ones. Neither pattern suggests that familiarity with the published networks drives accuracy.

Difficulty is non-monotone in bi-directed density. Very sparse graphs are mostly identifiable and very dense ones mostly non-identifiable, so errors concentrate at intermediate densities, where the verdict is genuinely contested. For FLASH on the main pool, false claims peak at 0.154 around 10 bi-directed edges and fall to 0 beyond 14 bi-directed edges.

Estimands. Among correctly identified instances, returned estimands are correct at 98.8–98.9% (FLASH 172/174, PRO 170/172; Sec. 4.3). Verification is limited to graphs with at most 9 vertices, leaving 41–42% of cases unchecked. GPT-5.5 estimands were not graded because its runner records verdicts without invoking the verifier.

Single runs are samples. Temperature zero did not produce deterministic outputs in Gemini models. Across 3 fresh responses for each of 200 instances, FLASH is non-unanimous on 4.5% and PRO on 2.5%, while majority voting yields no accuracy gain. FLASH's instability occurs entirely where a flip changes a correct refusal into a false claim. Thus, every reported figure from these endpoints is a sample rather than a constant, and 1-2 point gap fall within rerun variability (Appendix G.10).

5.2 WARRANTS ARE WITHHELD BY DEFAULT, NOT UNAVAILABLE

A correct verdict and a checkable justification are independent, and under our default prompt the justification is usually absent. Of FLASH's 291 correct refusals, *every one* is exactly 25 characters: the verdict line and nothing else. GPT-5.5 is identical: all 300 correct refusals are exactly 25 characters. PRO averages 472 characters, but its median is also 25. The distribution is bimodal, with PRO returning a bare verdict on 74.2% of its 283 correct refusals.

Asked directly, every model complies. When we re-run the same non-identifiable instances with a prompt that requests the structural reason, the silence disappears. None verdict-only reply survives in any model (0% across 883 correct refusals), and every reply names at least one graphical obstruction (100%). Mean lengths become 450.1, 501.9 and 437.3 characters. The shift is sharpest for PRO: its mean barely changes, but its distribution collapses, with the median rising from 25 to 502 and the 90th percentile falling from 1,933 to 603. The cited terms track the graph, with structures irrelevant to a non-identifiable query appearing near zero (Appendix G).

For one model, asking costs accuracy. FLASH's correct-refusal rate falls from 97.0% to 95.3% on the same 300 instances, with false claims rising from 9 to 14. PRO improves slightly, while GPT-5.5 remains exact. The shift is roughly 1.7× FLASH's measured flip rate (Appendix G.10), exceeding rerun variability but not by enough for a single run to establish the effect. We report it as suggestive.

The warrant is therefore withheld, not unavailable. Both silent models reason internally at length and articulate the obstruction on request, so default silence is a property of the interface rather than a model limitation. Our scope is precise: we measure what is *emitted*, not whether the emitted text reflects the computation that produced the verdict. The latter is the question studied by the faithfulness literature (Appendix B.4), which we do not address.

5.3 Accuracy Is a Poor Proxy for Soundness

Across graph sizes from 10 to 50 vertices, accuracy among the 3 models spans roughly 10 points, while the false-claim rate spans 17-fold (Figure 3). FLASH and PRO differ by 4 accuracy points but 2-fold in false claims; PRO and GPT-5.5 differ by 5 points but 7-fold. An accuracy leaderboard would present these systems as close competitors while concealing the property that distinguishes them. The comparison is like-for-like in instances, prompts, and parsing: every false claim and correct refusal for FLASH and GPT-5.5 was read from an explicit verdict line, with no fallback parsing (Appendix G).

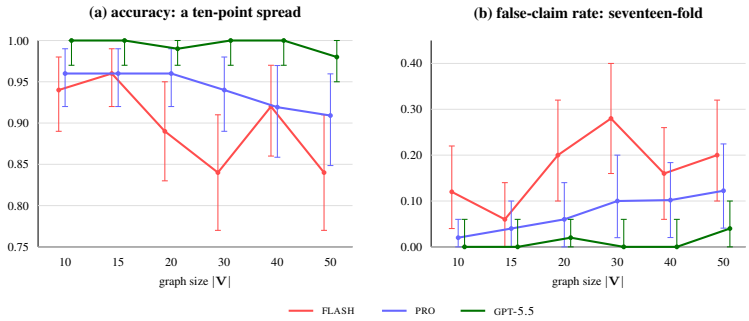


Figure 3: Accuracy and false-claim rate against graph size, three frontier models on identical instances and identical prompts. **(a)** Accuracy separates the models by about ten points and declines gently. **(b)** The false-claim rate separates them by a factor of 17 pooled.

Two conclusions follow. First, soundness degradation with graph size is not a property of frontier models. The degradation is pronounced in one family but absent in another, at sizes where FLASH loses up to 12 points of accuracy, so one system's behavior does not predict another's. Second, even the best model is not sound. A false claim on a non-identifiable query cannot be captured downstream, and GPT-5.5 asserted an answer to three provably unanswerable queries out of 300: small, but not zero, whereas a certified procedure is zero by construction.

5.4 Additional Results

On a published graph: We apply CertID to the causal diagram of Robins et al. (2000), verifying the authors' g-formula against exact ground truth and showing their identifying assumption is sufficient (Appendix H.4, Figure 5). **Truncation:** Excluding PRO's truncated main pool replies changes no comparison (Appendix G.5). **Small models:** Three open-weights models, from 0.5B to 3B parameters, perform at or below chance on edge classification (Appendix G.6). **Scale:** Appendix G.8 reports results by graph size with confidence intervals, and finds that larger graphs add mostly vertices irrelevant to the query. **Repair:** Without tools, 92.6–97.0% of proposed repairs are valid, and 96.8–99.3% of those are minimal. With ID as a tool, every repair returned by FLASH is valid and minimal, including on graphs too large to enumerate all minimal repairs (Appendix G.10).

6 Conclusion

CertID turns causal identification into an evaluation framework with certified labels and verifiable grades. Beyond deciding identifiability, its theoretical results characterize structural leakage, repair non-identifiable queries, and provide guarantees for numerical grading. Using this framework, we find that frontier reasoning models decide identifiability accurately even on unseen random graphs, yet often provide no justification unless asked. Most importantly, we show that accuracy is a poor proxy for soundness, and that soundness must therefore be measured directly.

AI USE STATEMENT

Following the categories in the ICLR 2027 AI Policy for Authors, we used generative AI tools for the following tasks with required disclosure: implementing methods, cleaning and reformatting data, and interpreting results. Among the recommended disclosure tasks, we used them for creating and editing software code, creating the artifact, drafting parts of the paper and editing it for readability, searching for and summarizing literature, and identifying relevant work. Portions of the manuscript were drafted with AI assistance and revised by the authors. We have reviewed all AI-assisted work, and we are explicit about the limits of that review. Every identifiability label was computed by two independently developed implementations of the ID algorithm, which agreed on every instance across $\sim 14.9M$ calls, and both were checked against a set of instances whose verdicts are stated in the published literature and transcribed here from rendered primary sources.

ETHICS STATEMENT

This work involves no human subjects, no personally identifying data and no newly collected data. The published Bayesian networks and causal diagrams are drawn from the peer-reviewed literature and cited. CERTID is a measurement instrument for formal causal identification, intended for evaluation and not for deployment. One finding concerns third-party software: Appendix F.4 reports two graphs on which a published causal inference library returns numerically incorrect estimands. We verified both against exact ground truth, and note there that the library's identifiability verdicts are correct in both cases, so the defect does not bear on its primary function. The API compute for the OpenAI experiments was from a personal account. The authors declare no competing interests arising from it.

REPRODUCIBILITY STATEMENT

The labeling procedure is deterministic given a graph and a query: every label is the output of the ID algorithm, computed independently by two implementations that agreed on every instance (Sec. 3.1, Appendix F). Instance generation, the leakage diagnostic and its remediation are specified in Sec. 3, with full detail in Appendix E. Grading is specified in Sec. 4.3, including the number of structural causal models drawn per instance, the numerical tolerance, and the two restrictions under which the comparison is made. Proofs of all stated results are in Appendix D. Model identifiers, pin quality and full revision hashes appear in Appendix F.5. Two of the three frontier endpoints are floating aliases that cannot be re-run against identical weights, and their verdicts are not stable under repetition (Appendix G.10). The open-weights results are pinned by revision hash and reproduce exactly. Code, benchmark instances, per-instance model responses and the summary files behind every reported number are available at `https://anonymous.4open.science/r/certid-D718`.

REFERENCES

Sina Akbari, Jalal Etesami, and Negar Kiyavash. Minimum cost intervention design for causal effect identification. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162, pp. 258–289, 2022.

Sina Akbari, Jalal Etesami, and Negar Kiyavash. Experimental design for causal effect identification. *arXiv preprint arXiv:2205.02232*, 2023.

Alexander Balke and Judea Pearl. Bounds on treatment effects from studies with imperfect compliance. *Journal of the American Statistical Association*, 92(439):1171–1176, 1997.

Ingo A Beinlich, Henri Jacques Suermondt, R Martin Chavez, and Gregory F Cooper. The alarm monitoring system: A case study with two probabilistic inference techniques for belief networks. In *AIME 89: Second European Conference on Artificial Intelligence in Medicine, London, August 29th–31st 1989. Proceedings*, pp. 247–256. Springer, 1989.

John Binder, Daphne Koller, Stuart Russell, and Keiji Kanazawa. Adaptive probabilistic networks with hidden variables. *Machine Learning*, 29(2):213–244, 1997.

Marcel Binz and Eric Schulz. Using cognitive psychology to understand gpt-3. *Proceedings of the National Academy of Sciences*, 120(6):e2218523120, 2023.

Zana Buçinca, Maja Barbara Malaya, and Krzysztof Z Gajos. To trust or to think: cognitive forcing functions can reduce overreliance on ai in ai-assisted decision-making. *Proceedings of the ACM on Human-computer Interaction*, 5(CSCW1):1–21, 2021.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han. Unveiling causal reasoning in large language models: Reality or mirage? *Advances in Neural Information Processing Systems*, 37:96640–96670, 2024.

Carlos Cinelli and Chad Hazlett. Making sense of sensitivity: Extending omitted variable bias. *Journal of the Royal Statistical Society: Series B*, 82(1):39–67, 2020.

Julian Coda-Forno, Marcel Binz, Jane X Wang, and Eric Schulz. Cogbench: a large language model walks into a psychology lab. *arXiv preprint arXiv:2402.18225*, 2024.

Sepehr Elahi, Sina Akbari, Jalal Etesami, Negar Kiyavash, and Patrick Thiran. Fast proxy experiment design for causal effect identification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Thilo Hagendorff, Ishita Dasgupta, Marcel Binz, Stephanie CY Chan, Andrew Lampinen, Jane X Wang, Zeynep Akata, and Eric Schulz. Machine psychology. *arXiv preprint arXiv:2303.13988*, 2023.

Paul He, Yinya Huang, Mrinmaya Sachan, and Zhijing Jin. Uncovering hidden correctness in LLM causal reasoning via symbolic verification. *arXiv preprint arXiv:2601.21210*, 2026.

Yimin Huang and Marco Valtorta. Pearl's calculus of intervention is complete. In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 217–224, 2006.

Amin Jaber, Jiji Zhang, and Elias Bareinboim. Causal identification under Markov equivalence: Completeness results. In *International Conference on Machine Learning (ICML)*, pp. 2981–2989, 2019.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, et al. CLADDER: Assessing causal reasoning in language models. *Advances in Neural Information Processing Systems*, 36: 31038–31065, 2023.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Emre Kiciman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality. *Transactions on Machine Learning Research*, 2023.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel J Bell. AbstentionBench: Reasoning LLMs fail on unanswerable questions. *Advances in Neural Information Processing Systems*, 38, 2026.

N Lambert, J Morrison, V Pyatkin, S Huang, H Ivison, F Brahman, LJV Miranda, A Liu, N Dziri, S Lyu, et al. 3: Pushing frontiers in open language model post-training. corr, abs/2411.15124, 2024. doi: 10.48550. *arXiv preprint ARXIV.2411.15124*, pp. 9, 2024.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. 2023.

Steffen L Lauritzen and David J Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

Donggyu Lee, Hyeok Yun, Meeyoung Cha, Sungwon Park, Sangyoon Park, and Jihee Kim. EconCausal: A context-aware economic reasoning benchmark for large language models. *arXiv preprint arXiv:2510.07231*, 2026.

Xiao Liu, Zirui Wu, Xueqing Wu, Pan Lu, Kai-Wei Chang, and Yansong Feng. Are LLMs capable of data-based statistical and causal reasoning? Benchmarking advanced quantitative reasoning with data. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9215–9235, 2024.

Charles F Manski. Nonparametric bounds on treatment effects. *The American Economic Review*, 80(2):319–323, 1990.

Agnieszka Onisko. Probabilistic causal models in medicine: Application to diagnosis of liver disorders. In *Ph. D. dissertation, Inst. Biocybern. Biomed. Eng., Polish Academy Sci., Warsaw, Poland*, 2003.

Raja Parasuraman and Victor Riley. Humans and automation: Use, misuse, disuse, abuse. *Human factors*, 39(2):230–253, 1997.

Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, Cambridge, UK, 2nd edition, 2009.

Emilija Perković, Johannes Textor, Markus Kalisch, and Marloes H Maathuis. Complete graphical characterization and construction of adjustment sets in Markov equivalence classes of ancestral graphs. *Journal of Machine Learning Research*, 18(220):1–62, 2018.

James M Robins, Miguel Angel Hernan, and Babette Brumback. Marginal structural models and causal inference in epidemiology. *Epidemiology*, 11(5):550–560, 2000.

Paul R Rosenbaum. *Observational Studies*. Springer, 2nd edition, 2002.

Pål Christie Ryalen, Mats Julius Stensrud, and Kjetil Røysland. On causal inference with marked point process data. *arXiv preprint arXiv:2604.12977*, 2026.

Ayush Sawarni, Jiyuan Tan, and Vasilis Syrgkanis. CausalReasoningBenchmark: A real-world benchmark for disentangled evaluation of causal identification and estimation. *arXiv preprint arXiv:2602.20571*, 2026.

Max Schemmer, Niklas Kuehl, Carina Benz, Andrea Bartos, and Gerhard Satzger. Appropriate reliance on ai advice: Conceptualization and the effect of explanations. In *Proceedings of the 28th international conference on intelligent user interfaces*, pp. 410–422, 2023.

Jacob T Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM (JACM)*, 27(4):701–717, 1980.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Ilya Shpitser and Judea Pearl. Identification of conditional interventional distributions. In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 437–444, 2006a.

Ilya Shpitser and Judea Pearl. Identification of joint interventional distributions in recursive semi-Markovian causal models. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI)*, pp. 1219–1226, 2006b.

Ilya Shpitser and Judea Pearl. Complete identification methods for the causal hierarchy. *Journal of Machine Learning Research*, 9:1941–1979, 2008.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. *Advances in neural information processing systems*, 35:9460–9471, 2022.

David J Spiegelhalter. Learning in probabilistic expert systems. *Bayesian statistics*, 4:447–465, 1992.

Peter WG Tennant, Eleanor J Murray, Kellyn F Arnold, Laurie Berrie, Matthew P Fox, Sarah C Gadd, Wendy J Harrison, Claire Keeble, Lynsie R Ranker, Johannes Textor, et al. Use of directed acyclic graphs (dags) to identify confounders in applied health research: review and recommendations. *International journal of epidemiology*, 50(2):620–632, 2021.

Jin Tian and Judea Pearl. A general identification condition for causal effects. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, pp. 567–573, 2002.

Santtu Tikka and Juha Karvanen. Identifying causal effects with the R package causaleffect. *Journal of Statistical Software*, 76(12):1–30, 2017a.

Santtu Tikka and Juha Karvanen. Simplifying probabilistic expressions in causal inference. *Journal of Machine Learning Research*, 18(36):1–30, 2017b.

Santtu Tikka and Juha Karvanen. Enhancing identification of causal effects by pruning. *Journal of Machine Learning Research*, 18(194):1–23, 2018.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023.

Tyler J VanderWeele and Peng Ding. Sensitivity analysis in observational research: Introducing the E-value. *Annals of Internal Medicine*, 167(4):268–274, 2017.

Thomas S Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Probabilistic and causal inference: The works of Judea Pearl*, pp. 221–236. 2022.

Zeyu Wang. CausalBench: A comprehensive benchmark for evaluating causal reasoning capabilities of large language models. In *Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10)*, pp. 143–151, 2024.

Ziming Wang, Yinghua Yao, Changwu Huang, Ke Tang, and Xin Yao. CASE: Causal alignment and structural enforcement for improving chain-of-thought faithfulness. *arXiv preprint arXiv:2607.18820*, 2026.

Xingyu Wu, Kui Yu, Jibin Wu, and Kay Chen Tan. LLM cannot discover causality, and should be restricted to non-decisional support in causal discovery. *arXiv preprint arXiv:2506.00844*, 2025.

Zhi Xu and Yun Fu. NoisyCausal: A benchmark for evaluating causal reasoning under structured noise. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 39500–39513, 2026.

Linying Yang, Vik Shirvaikar Wang, Ahmed Alaa, and David Blei. A critical review of causal reasoning benchmarks for large language models. *arXiv preprint arXiv:2407.08029*, 2024.

Matej Zečević, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. Causal parrots: Large language models may talk causality but are not causal. *Transactions on Machine Learning Research*, 2023.

Yizhuo Zhang, Heng Wang, Shangbin Feng, Zhaoxuan Tan, Xiaochuang Han, Tianxing He, and Yulia Tsvetkov. Can LLM graph reasoning generalize beyond pattern memorization? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 2289–2305, 2024.

Richard Zippel. Probabilistic algorithms for sparse polynomials. In *International symposium on symbolic and algebraic manipulation*, pp. 216–226. Springer, 1979.

Appendix

A Notation and Formal Definitions **17**

A.1 Notation 17

A.2 C-components, C-forests and hedges 17

B Extended Related Work **18**

B.1 Identification given a graph 18

B.2 When identification fails 18

B.3 Benchmarks for causal reasoning 18

B.4 Chain-of-thought faithfulness 19

B.5 Language models and causal reasoning 19

C Properties of the Certification Procedure **19**

C.1 Structure and cost of the restoring set 19

C.2 Scope of the grading guarantee 20

C.3 What the reported numbers license 21

D Proofs **21**

D.1 Monotonicity and minimization 21

D.2 Hedge destruction and its consequences 22

D.3 Grading 23

D.4 Metrics 23

E Structural Leakage in Semi-Synthetic Construction **24**

E.1 Features and classifier 24

E.2 The leak 24

E.3 The construction 24

E.4 The leakage that remains 25

F Verification of the Labeling Procedure **25**

F.1 Canonical known-answer set 25

F.2 Two implementations of ID 25

F.3 The verifier 26

F.4 Five defects in a published causal-inference library 26

F.5 Model identities 27

G Evaluation Protocol and Additional Results **27**

G.1 Published networks 27

G.2 Output budgets 27

G.3 Graph serialization 28
G.4 How a verdict is read 28
G.5 Truncation 28
G.6 Small open-weights models in full 29
G.7 Scale grid, numerically 29
G.8 Accuracy and soundness under scale, in full 29
G.9 Repair and enumeration in full 30
G.10 Reproducibility at temperature zero 31

H Worked Examples **31**
H.1 A certified label: a non-identifiable instance and its frontier 31
H.2 A false claim by the strongest model 32
H.3 A verified grade: an estimand that is proper and wrong 33
H.4 A published graph: Robins, Hernán and Brumback (2000) 34

I Discussion **35**
I.1 What the results do and do not say about causal competence 35
I.2 Using these results in practice 35
I.3 Applications of this work 36
I.4 Limitations 37

A Notation and Formal Definitions

This appendix supports the body in two ways. Appendix A.1 collects every symbol, so a reader can check a definition without searching Sections 2–4. Appendix A.2 states the graphical characterization of non-identifiability. The body uses it only through ID, except in Theorem 1 and its proof. The hedge definitions are prior work.

A.1 Notation

Table 3 collects every symbol used in the paper. Uppercase denotes random variables, lowercase their realizations, and boldface sets of variables. Graph-theoretic notation follows Pearl (2009), and identification notation follows Shpitser & Pearl (2006b).

Table 3: Symbols used in the paper, with the section in which each is introduced.

Symbol	Meaning	Introduced
graphs and models		
G	acyclic directed mixed graph over $\mathbf{V}$	Sec. 2
$\mathbf{V}$	observed variables of G	Sec. 2
$\mathbf{X}, \mathbf{Y}$	disjoint treatment and outcome sets	Sec. 2
$B(G)$	bi-directed edges of G	Sec. 2
$\mathrm{An}(\mathbf{Y})$	ancestors of $\mathbf{Y}$	Sec. 2
$G_{\overline{\mathbf{X}}}$	G with edges into $\mathbf{X}$ removed	App. A.2
$\mathcal{M}$, $\mathcal{M}(G)$	an SCM; the SCMs whose dependencies lie in G	Sec. 2
$P(\mathbf{V})$	observational distribution	Sec. 2
θ	parameters of a discrete SCM	Thm. 2
$F^{\leftrightarrow}$	bi-directed edges of a subgraph F	App. D
identification		
$Q = P_{\mathbf{x}}(\mathbf{y})$	query $P(\mathbf{y} \mid \mathrm{do}(\mathbf{x}))$	Sec. 2
$\mathrm{ID}(G, Q)$	identification algorithm; returns ϕ or FAIL	Sec. 2
ϕ	a closed-form estimand	Sec. 2
$\mathbf{R}$	root set of a C-forest	App. A.2
$\mathcal{H}(G, Q)$	hedges for Q in G over all $\mathbf{X}' \subseteq \mathbf{X}$, $\mathbf{Y}' \subseteq \mathbf{Y}$	Thm. 1
problem and evaluation		
ℓ	certified label $\mathrm{ID}(G, Q)$	Def. 3
$r = (v, \hat{\phi})$	response: verdict and returned estimand	Def. 5
k, τ	SCM draws per instance; numerical tolerance	Def. 4
$\mathcal{S}, \mathcal{I}, \mathcal{N}$	scored, identifiable and non-identifiable instances	Sec. 4.4
construction and repair		
$p_{\mathrm{dir}}, p_{\mathrm{bi}}$	directed and bi-directed edge probabilities	Def. 6
D, L	a published DAG and the latent set hidden from it	Def. 7
$A \subseteq B(G)$	augmentation: bi-directed edges assumed absent	Sec. 3.3
$G \ominus A$	G with the edges of A deleted	Sec. 3.3
$\mathcal{A}(G, Q)$	augmentations under which Q becomes identifiable	Sec. 3.3
$\mathcal{F}(G, Q)$	$\subseteq$-minimal elements of $\mathcal{A}(G, Q)$	Sec. 3.3

A.2 C-components, C-forests and hedges

The body needs hedges only for Theorem 1, which states when an augmentation destroys a hedge. Its proof uses the fact that most conditions in the definition below survive the deletion of a bi-directed edge. The three definitions and the criterion are stated here in the form given by Shpitser & Pearl (2006b).

Definition 10 (C-component; Tian & Pearl, 2002)**.** A set $\mathbf{C} \subseteq \mathbf{V}$ is a *C-component* of G if every pair of vertices in $\mathbf{C}$ is connected by a path consisting entirely of bi-directed edges.

Definition 11 (C-forest; Shpitser & Pearl, 2006b)**.** Let F be a subgraph of G, and let its root set $\mathbf{R}$ be the vertices of F that have no children in F. F is an $\mathbf{R}$*-rooted C-forest* if the bi-directed edges of F connect all its vertices and every vertex of F has at most one child in F.

Definition 12 (Hedge; Shpitser & Pearl, 2006b)**.** Let F and F' be $\mathbf{R}$-rooted C-forests with $F' \subseteq F$, $F \cap \mathbf{X} \neq \emptyset$, $F' \cap \mathbf{X} = \emptyset$, and $F \subseteq \mathrm{An}(\mathbf{Y})_{G_{\overline{\mathbf{X}}}}$. Then F, F' form a *hedge* for $P_{\mathbf{x}}(\mathbf{y})$ in G.

Hedge criterion. *$P_{\mathbf{x}}(\mathbf{y})$ is identifiable from $P(\mathbf{V})$ in G if and only if there is no hedge for $P_{\mathbf{x}'}(\mathbf{y}')$ in G for any $\mathbf{X}' \subseteq \mathbf{X}$ and $\mathbf{Y}' \subseteq \mathbf{Y}$ (Shpitser & Pearl, 2006b).*

The criterion quantifies over all subsets of $\mathbf{X}$ and $\mathbf{Y}$, so a single graph may contain many hedges and enumerating them is expensive. Theorem 1 therefore takes the hedges as given.

B Extended Related Work

This appendix places our work in five literatures: identification given a graph, what to do when identification fails, benchmarks for causal reasoning, chain-of-thought faithfulness, and studies of causal reasoning in language models.

B.1 Identification given a graph

Given a causal graph, the question of which interventional quantities are recoverable from observational data is settled. Do-calculus provides the rewrite rules (Pearl, 1995) and is complete for interventional identification (Huang & Valtorta, 2006). ID decides the question constructively and returns a hedge on failure (Tian & Pearl, 2002; Shpitser & Pearl, 2006b; 2008), with extensions to conditional effects (Shpitser & Pearl, 2006a) and to Markov equivalence classes (Jaber et al., 2019). Returned expressions can be simplified and their graphs pruned (Tikka & Karvanen, 2017b; 2018), and valid adjustment sets are completely characterized (Perković et al., 2018). We use these results without modification. The completeness of ID makes our labels theorems, and its polynomial running time lets us generate instances at arbitrary scale.

B.2 When identification fails

Two responses exist. The first repairs the graph. Akbari et al. (2022) pose minimum-cost intervention design, show it to be NP-hard, and give an exact algorithm and a logarithmic-factor approximation by casting it as a minimum hitting set over hedges: an intervention set restores identifiability exactly when it hits every hedge. Elahi et al. (2024) show the number of minimal hedges can be exponential and recast the problem as weighted MAX-SAT, an integer program, and submodular maximization. Akbari et al. (2023) extend the problem to experimental design. The second response accepts the failure and quantifies its consequences: partial identification returns bounds (Manski, 1990; Balke & Pearl, 1997), and sensitivity analysis describes how conclusions erode as an assumption weakens (Rosenbaum, 2002; Cinelli & Hazlett, 2020; VanderWeele & Ding, 2017). Our repair task differs from the first response in what it changes. Deleting a bi-directed edge asserts that a confounder is absent, while an intervention removes the confounder's influence by setting a variable directly. Theorem 1 shows that this changes what restores identifiability: an assumption set must cut the bi-directed edges of every hedge, where an intervention set must hit its vertices. We do not study the resulting optimization problem; we use the characterization to score repair responses exactly.

B.3 Benchmarks for causal reasoning

CLADDER verbalizes symbolic causal problems across the three rungs of the ladder of causation and solves each with an inference engine to obtain ground truth (Jin et al., 2023). QRDATA pairs causal questions with real data tables (Liu et al., 2024); CAUSALBENCH probes each scenario from several directions to reduce success by chance (Wang, 2024), and NOISYCAUSAL generates instances from ground-truth graphs and injects structured noise, including latent confounders (Xu & Fu, 2026). In each case the ground truth is a value computed from a fully specified model, and every question posed is answerable by construction. ECONCAUSAL evaluates context-dependent causal claims drawn from empirical economics and reports a finding adjacent to ours: offered an explicit *Unknown* option with no identifying context, models select that option under 10% of the time and commit to a directional sign (Lee et al., 2026). Their unanswerable cases arise from underspecification, while ours are non-identifiable as a theorem. That difference lets us measure a false-claim rate against ground truth, where they measure a tendency to over-commit. On grading, He et al. (2026)

address the same problem by checking symbolically whether a generated expression is derivable from the graph under do-calculus. We grade numerically instead, by evaluating the expression on an SCM whose interventional distribution is known exactly. Numerical grading accepts a correct expression written in any form. ABSTENTIONBENCH evaluates when models should decline to answer (Kirichenko et al., 2026), with unanswerable instances fixed by annotation where ours are decided by a complete algorithm.

Two recent lines converge on our motivation from different directions. Yang et al. (2024) survey causal reasoning benchmarks and observe that many can be solved by retrieving domain knowledge, which questions whether those benchmarks measure what they claim. The random family of Sec. 3.1 removes that route by construction. Closer still, CRB (Sawarni et al., 2026) argues that scoring a single numerical output conflates identification with estimation, and separates the two by requiring a structured research design alongside a point estimate on real datasets. We share the diagnosis and differ in two respects: what identification means, and where the ground truth comes from. Their specifications name a design (instrumental variables, difference-in-differences, regression discontinuity) and a control set, curated from published papers. Ours is graphical identifiability, decided by ID and graded by numerical evaluation. Their labels are curated; ours are computed. The two are complementary, and neither subsumes the other.

B.4 CHAIN-OF-THOUGHT FAITHFULNESS

A parallel literature asks whether a generated reasoning trace reflects the computation that produced the answer. Models fail to verbalize cues that demonstrably determined their output (Turpin et al., 2023; Chen et al., 2025). Faithfulness is measured by perturbing the trace and observing whether the answer changes (Lanham et al., 2023), and methods such as CASE intervene on training and inference to force the answer through the trace (Wang et al., 2026). All of this presupposes that a trace exists. Sec. 5.2 reports that on this task a trace frequently does not exist. Under a default prompt, two of three frontier models return every correct refusal as a bare verdict, so the question of faithfulness does not arise until a trace is requested.

B.5 LANGUAGE MODELS AND CAUSAL REASONING

Behavioral studies attribute measured causal competence to knowledge learned from text, attached to variable names, and find little evidence of structural analysis (Kiciman et al., 2023; Zečević et al., 2023). Chi et al. (2024) report sharp degradation on corpora published after the training cutoff. Zhang et al. (2024) find that graph reasoning fails to transfer under controlled distribution shift, and Wu et al. (2025) argue that language models should be confined to non-decisional roles in causal discovery. Sec. 5.1 reports results that sit uneasily with the strongest form of these claims, measured on a task where neither variable names nor memorized structure is available.

C PROPERTIES OF THE CERTIFICATION PROCEDURE

This appendix builds on three results from the body. Identifiability is monotone under edge deletion (Proposition 1). Grading refutes a wrong estimand almost surely (Theorem 2). An augmentation restores identifiability exactly when the augmentation destroys every hedge (Theorem 1). This appendix records what follows from them. The consequences answer four questions: how large the frontier can be and which edges it uses, how expensive membership is, how much of the grading guarantee survives our implementation's approximations, and what the reported accuracy and leakage license. Nothing here is required to follow the body, and all proofs are deferred to Appendix D.

C.1 STRUCTURE AND COST OF THE RESTORING SET

Proposition 1 makes $\mathcal{A}(G, Q)$ an up-set, so its minimal elements cannot contain one another.

Corollary 1 (Antichain bound)**.** *$\mathcal{F}(G, Q)$ is an anti-chain in the Boolean lattice on $B(G)$, so with $m = |B(G)|$,*

$$|\mathcal{F}(G, Q)| \;\leq\; \binom{m}{\lfloor m/2 \rfloor}.$$

The bound is exponential in m and is attained by the family of all $\lfloor m/2 \rfloor$-subsets, so nothing about the lattice structure prevents the frontier from being large. Exhaustive enumeration becomes infeasible in Table 13 for a different reason: it searches all 2^m subsets of $B(G)$, and m grows with the graph. The oracle-guided search queries only the subsets it needs. Enumeration pays for the whole anti-chain; a search that queries only the subsets it needs does not. Relatedly, Elahi et al. (2024) exhibit graphs whose minimal hedges are exponentially many.

Theorem 1 reduces membership to a connectivity question once the hedges are known.

Corollary 2 (Membership without calling ID)**.** *Given $\mathcal{H}(G, Q)$, deciding $A \in \mathcal{A}(G, Q)$ costs $O(|\mathcal{H}| \cdot (|\mathbf{V}| + |B(G)|))$ and no call to* ID.

We do not use this corollary in our evaluation, because obtaining $\mathcal{H}(G, Q)$ is itself the expensive step: ID returns one witness, not all hedges. The corollary matters for a different reason. It locates where the cost of re[Membership without an oracle call]pair actually sits. Deciding whether a proposed augmentation works is cheap given the obstructions; finding the obstructions is not.

Remark 1 (Covering by cuts, not by elements). By Theorem 1, $A \in \mathcal{A}(G, Q)$ exactly when A contains a disconnecting set of $F^{\leftrightarrow}$ or of $F'^{\leftrightarrow}$ for every $(F, F') \in \mathcal{H}(G, Q)$. The covering objects are therefore edge cuts, not single edges. That distinguishes our formulation from the intervention design of Akbari et al. (2022), where an intervention set restores identifiability exactly when the set *hits* every hedge.

In one case, the edge formulation also becomes a hitting-set problem.

Corollary 3 (Reduction to hitting set in the tree case)**.** *If the bidirected structure of every hedge in $\mathcal{H}(G, Q)$ is a tree, then $A \in \mathcal{A}(G, Q)$ exactly when A meets $F^{\leftrightarrow}$ for every $(F, F') \in \mathcal{H}(G, Q)$. Finding a minimum-cardinality restoring augmentation is then an instance of minimum hitting set over the family $\{F^{\leftrightarrow}\}$.*

We state this as a reduction and not as a hardness result, and the distinction matters. Corollary 3 exhibits our problem as an instance of hitting set, which bounds it from above and says nothing about its difficulty. Establishing NP-hardness requires the opposite direction: a construction sending an arbitrary hitting-set instance to a graph whose hedge family realizes it. Akbari et al. (2022) give such a construction for the vertex formulation, where an intervention set restores identifiability exactly when it hits every hedge. We have not carried it over to edge deletion and make no hardness claim for the edge formulation.

Two further consequences of Theorem 1 locate the edges a minimal repair can use and bound its size.

Corollary 4 (Minimal repairs lie inside $\mathrm{An}(\mathbf{Y})$)**.** *No element of $\mathcal{F}(G, Q)$ contains a bi-directed edge with an endpoint outside $\mathrm{An}(\mathbf{Y})$.*

Enlarging the graph outside $\mathrm{An}(\mathbf{Y})$ therefore adds no candidate edge to any minimal repair.

For a subgraph F, let $\lambda(F)$ be the least number of edges of $F^{\leftrightarrow}$ whose deletion leaves $V(F)$ disconnected, with $\lambda(F) = \infty$ when F has a single vertex.

Corollary 5 (Lower bound on repair size)**.** *Every $A \in \mathcal{A}(G, Q)$ satisfies $|A| \geq \min\{\lambda(F), \lambda(F')\}$ for each $(F, F') \in \mathcal{H}(G, Q)$. Over a family of hedges whose sets $F^{\leftrightarrow}$ are pairwise disjoint, the bounds add.*

The bound needs no enumeration. Each λ is a minimum edge cut, computable in polynomial time, and the bound holds for any known hedge, including the one ID returns on failure.

C.2 Scope of the grading guarantee

Theorem 2 is stated with $\tau = 0$ and exact arithmetic. Our implementation uses neither. Three consequences delimit what survives.

Corollary 6 (One draw suffices in exact arithmetic)**.** *Under the hypotheses of Theorem 2 with $\tau = 0$, an estimand that disagrees with Q on the parametrized family is refuted by a single draw with probability one. Drawing $k > 1$ SCMs therefore adds nothing to the guarantee.*

In exact arithmetic, one draw therefore suffices. We draw $k = 2$ because of floating-point arithmetic: with $\tau > 0$, a wrong estimand can agree with the target within tolerance on one draw by coincidence, and a second draw makes that unlikely. The second draw adds robustness to finite precision, not certainty about the estimand.

Remark 2 (The guarantee degrades continuously in τ). Let $D(\theta) = \hat{\phi}(P_\theta(\mathbf{V})) - Q_\theta$. The sets $\{\theta : |D(\theta)| \leq \tau\}$ decrease as $\tau \downarrow 0$ and intersect in $\{\theta : D(\theta) = 0\}$, which has measure zero by Theorem 2. Their measure therefore tends to zero with τ: a wrong estimand escapes refutation on a positive but vanishing fraction of parameter space.

Remark 3 (The comparison certifies a slice). Our implementation compares at one intervention value (Sec. 4.3), so Theorem 2 certifies agreement on $P(\mathbf{y} \mid \mathrm{do}(x))$ at that value. Certifying the full interventional distribution requires the comparison at each value in the domain of $\mathbf{X}$, multiplying the cost of the enumeration by $|\mathrm{dom}(\mathbf{X})|$. That strengthening is the obvious one, and we did not run it.

Remark 4 (A refutation is a certificate). A refutation exhibits a parameter θ with $|\hat{\phi}(P_\theta(\mathbf{V})) - Q_\theta| > \tau$. Given θ, anyone can recompute both quantities, in exact rational arithmetic if desired, and confirm the refutation without trusting our implementation. Verification has no such certificate; its strength is Theorem 2(ii).

C.3 What the reported numbers license

Corollary 7 (Accuracy does not determine the false-claim rate)**.** *Let a pool have $|\mathcal{I}| = |\mathcal{N}|$. Let a system have false-claim rate f and miss rate m, the fraction of $\mathcal{I}$ answered with a miss. Its accuracy is $a = 1 - (f + m)/2$. Hence, for $a \geq 1/2$, the false-claim rate satisfies $0 \leq f \leq 2(1 - a)$, and both bounds are attained.*

At $a = 0.98$, the false-claim rate can lie anywhere in $[0, 0.04]$; at $a = 0.90$, anywhere in $[0, 0.20]$. Accuracy therefore cannot rank systems by soundness, and Sec. 5.3 observes this spread in practice.

Remark 5 (Reported leakage bounds a classifier, not the graph). Leakage (Definition 8) is the AUC of one classifier. A better classifier on graph features alone may separate real from spurious edges further. Exceeding the reported leakage is therefore necessary, and not sufficient, for a system to be credited with using the variable names.

D Proofs

This appendix proves every result stated in the body and in Appendix C, in the order they appear. Theorem 1 and its consequences use the definitions of Appendix A.2.

D.1 Monotonicity and minimization

Proof of Proposition 1. Let G' be G with any set of edges deleted. By definition, $\mathcal{M}(G)$ contains every SCM whose structural dependencies are contained in G. Every dependency permitted by G' is permitted by G, so $\mathcal{M}(G') \subseteq \mathcal{M}(G)$. If Q is identifiable in G there is a functional f with $Q_{\mathcal{M}} = f(P_{\mathcal{M}}(\mathbf{V}))$ for every $\mathcal{M} \in \mathcal{M}(G)$; the same f serves every $\mathcal{M} \in \mathcal{M}(G')$, so Q is identifiable in G'.

For upward closure, let $A \in \mathcal{A}(G, Q)$ and $A \subseteq A'$. Writing $G \ominus A' = (G \ominus A) \ominus (A' \setminus A)$ and applying the above to $G \ominus A$ gives $A' \in \mathcal{A}(G, Q)$.

For non-emptiness, $G \ominus B(G)$ has no bi-directed edges, so every SCM it represents is Markovian and every interventional distribution is identifiable by truncated factorization (Pearl, 2009). Hence $B(G) \in \mathcal{A}(G, Q)$, the set is non-empty, and since it is a finite up-set it is the upward closure of its minimal elements $\mathcal{F}(G, Q)$. □

Proof of Lemma 1. Write $A_0 = A$ and let A_i be the set after the ith element has been considered, so A_i is A_{i-1} with that element deleted when the result lies in $\mathcal{A}(G, Q)$ and equal to A_{i-1} otherwise. Every A_i lies in $\mathcal{A}(G, Q)$ by construction, so the result $A^\star = A_{|A|}$ does.

Suppose $A^\star \notin \mathcal{F}(G, Q)$. Then some $A'' \subsetneq A^\star$ lies in $\mathcal{A}(G, Q)$; choose $e \in A^\star \setminus A''$. Since $A'' \subseteq A^\star \setminus \{e\}$, upward closure gives $A^\star \setminus \{e\} \in \mathcal{A}(G, Q)$. Let i be the step at which e was

considered. Because elements are only ever removed, $A^\star \subseteq A_{i-1}$, so $A^\star \setminus \{e\} \subseteq A_{i-1} \setminus \{e\}$ and upward closure gives $A_{i-1} \setminus \{e\} \in \mathcal{A}(G, Q)$. The rule would then have deleted e, contradicting $e \in A^\star$. Each element is tested once, so the procedure makes exactly $|A|$ calls to ID. □

D.2 HEDGE DESTRUCTION AND ITS CONSEQUENCES

Theorem 1 rests on the fact that deleting bi-directed edges cannot create a hedge, which we state separately because both directions of the theorem use it.

Lemma 2 (Hedge persistence)**.** *Let $A \subseteq B(G)$. If (F, F') is a hedge for $P_{\mathbf{x}'}(\mathbf{y}')$ in $G \ominus A$, it is a hedge for the same query in G.*

Proof. The vertices of F are joined by bi-directed paths in $G \ominus A$, whose bi-directed edges are a subset of those of G, so they remain a C-component in G; the same holds for F'. The forest condition and the root set $\mathbf{R}$ are determined by the directed edges of F, which the two graphs share. The conditions $F' \subseteq F$, $F \cap \mathbf{X}' \neq \emptyset$ and $F' \cap \mathbf{X}' = \emptyset$ involve vertex sets only. Finally, ancestry is a directed notion and $G \ominus A$ and G have the same directed edges, so $\mathrm{An}(\mathbf{Y}')_{(G \ominus A)_{\overline{\mathbf{X}'}}} = \mathrm{An}(\mathbf{Y}')_{G_{\overline{\mathbf{X}'}}}$ and the containment is preserved. □

Proof of Theorem 1. Write $F^{\leftrightarrow}$ for the bi-directed edges of F.

Second clause first. Since $G \ominus A$ has a subset of the edges of G, F and F' are also subgraphs of G, and their bi-directed edges still connect their vertices. Every condition in the definition of a hedge except the two C-component conditions depends only on vertices and directed edges, so those conditions hold for $(F \setminus A, F' \setminus A)$ exactly as for (F, F'). The C-component conditions hold if and only if $F^{\leftrightarrow} \setminus A$ connects $V(F)$ and $F'^{\leftrightarrow} \setminus A$ connects $V(F')$. Hence $(F \setminus A, F' \setminus A)$ fails to be a hedge exactly when deleting A disconnects $F^{\leftrightarrow}$ or $F'^{\leftrightarrow}$.

First clause. If $A \in \mathcal{A}(G, Q)$, then Q is identifiable in $G \ominus A$, and the hedge criterion gives $\mathcal{H}(G \ominus A, Q) = \emptyset$. In particular, no pair $(F \setminus A, F' \setminus A)$ is a hedge in $G \ominus A$, so A destroys every element of $\mathcal{H}(G, Q)$. Conversely, suppose A destroys every element of $\mathcal{H}(G, Q)$, and let $(F, F') \in \mathcal{H}(G \ominus A, Q)$. By Lemma 2, $(F, F') \in \mathcal{H}(G, Q)$. Since F and F' are subgraphs of $G \ominus A$, neither contains an edge of A, so $(F \setminus A, F' \setminus A) = (F, F')$, which is a hedge in $G \ominus A$. This contradicts A destroying (F, F'). Hence $\mathcal{H}(G \ominus A, Q) = \emptyset$, and the hedge criterion gives identifiability. □

Proof of Corollary 1. If $A, A' \in \mathcal{F}(G, Q)$ with $A \subsetneq A'$ then A' is not $\subseteq$-minimal in $\mathcal{A}(G, Q)$, a contradiction; so no element of $\mathcal{F}(G, Q)$ properly contains another and the family is an antichain in the Boolean lattice on $B(G)$. Sperner's theorem bounds the size of an antichain on an m-element ground set by $\binom{m}{\lfloor m/2 \rfloor}$. □

Proof of Corollary 2. By Theorem 1, $A \in \mathcal{A}(G, Q)$ exactly when, for every $(F, F') \in \mathcal{H}(G, Q)$, deleting A disconnects the bi-directed edges of F or of F'. Each test is a connectivity query on a subgraph of G and runs in $O(|\mathbf{V}| + |B(G)|)$ by breadth-first search; there are two per hedge. □

Proof of Corollary 3. Suppose the bi-directed structure of every F in $\mathcal{H}(G, Q)$ is a tree. Removing any edge of a tree disconnects it, so A disconnects $F^{\leftrightarrow}$ exactly when A meets its edge set. Since $F' \subseteq F$, we have $F'^{\leftrightarrow} \subseteq F^{\leftrightarrow}$, so if A disconnects $F'^{\leftrightarrow}$, it meets $F^{\leftrightarrow}$. By Theorem 1, A destroys (F, F') exactly when it meets the bi-directed edge set of F, and $\mathcal{A}(G, Q)$ is the family of sets hitting every set in $\{F^{\leftrightarrow} : (F, F') \in \mathcal{H}(G, Q)\}$. Minimizing $|A|$ over that family is an instance of minimum hitting set. □

Proof of Corollary 4. Let $(F, F') \in \mathcal{H}(G, Q)$ be a hedge for $P_{\mathbf{x}'}(\mathbf{y}')$. By Definition 12, the roots of F are ancestors of $\mathbf{Y}' \subseteq \mathbf{Y}$. Every vertex of F is an ancestor of a root: each vertex has at most one child in F, so following children in the acyclic F ends at a root. Hence both endpoints of every edge in $F^{\leftrightarrow}$ and $F'^{\leftrightarrow}$ lie in $\mathrm{An}(\mathbf{Y})$.

Let $A \in \mathcal{F}(G, Q)$, and suppose $e \in A$ has an endpoint outside $\mathrm{An}(\mathbf{Y})$. Then e lies in no $F^{\leftrightarrow}$ or $F'^{\leftrightarrow}$. Whether deleting A disconnects $F^{\leftrightarrow}$ depends only on $A \cap F^{\leftrightarrow} = (A \setminus \{e\}) \cap F^{\leftrightarrow}$, and

likewise for F'. By Theorem 1, $A \setminus \{e\}$ therefore destroys every hedge that A destroys, which is all of $\mathcal{H}(G, Q)$, so $A \setminus \{e\} \in \mathcal{A}(G, Q)$. This contradicts the minimality of A. □

Proof of Corollary 5. By Theorem 1, A destroys (F, F'), so deleting A disconnects $F^{\leftrightarrow}$ or $F'^{\leftrightarrow}$. In the first case, $A \cap F^{\leftrightarrow}$ is a set of edges of $F^{\leftrightarrow}$ whose deletion disconnects $V(F)$, so $|A| \geq |A \cap F^{\leftrightarrow}| \geq \lambda(F)$. The second case gives $|A| \geq \lambda(F')$ in the same way.

For hedges $(F_1, F_1'), \ldots, (F_r, F_r')$ with pairwise disjoint $F_i^{\leftrightarrow}$, apply the same argument to each. Since $F_i'^{\leftrightarrow} \subseteq F_i^{\leftrightarrow}$, it gives $|A \cap F_i^{\leftrightarrow}| \geq \min\{\lambda(F_i), \lambda(F_i')\}$. Summing over the disjoint sets gives $|A| \geq \sum_i \min\{\lambda(F_i), \lambda(F_i')\}$. □

D.3 GRADING

Theorem 2 is stated relative to the family of SCMs the verifier draws from, so we fix that family first: discrete SCMs over $\mathbf{V}$ whose structural dependencies realize G, with one latent variable per bi-directed edge and a fixed number of levels per variable, parametrized by their free conditional probability table entries. Each $\theta \in \Theta$ determines a model $\mathcal{M}_\theta \in \mathcal{M}(G)$, so Θ indexes a subfamily of $\mathcal{M}(G)$ and not all of it. Say that $\hat{\phi}$ *denotes* Q on a class if $\hat{\phi}(P_{\mathcal{M}}(\mathbf{V})) = Q_{\mathcal{M}}$ for every model in that class.

Proof of Theorem 2. *(i)* Suppose the procedure refutes, so $\hat{\phi}(P_\theta(\mathbf{V})) \neq Q_\theta$ for some drawn θ. Since $\mathcal{M}_\theta \in \mathcal{M}(G)$, this is a single model of G on which $\hat{\phi}$ and Q disagree, so $\hat{\phi}$ does not denote Q on $\mathcal{M}(G)$.

(ii) Write $D(\theta) = \hat{\phi}(P_\theta(\mathbf{V})) - Q_\theta$. In a discrete SCM with latents materialized, $P_\theta(\mathbf{V})$ is obtained by summing products of conditional probability table entries over the latent configurations, so each of its entries is a polynomial in θ; Q_θ is obtained the same way from the truncated factorization and is likewise polynomial. The parsed form of $\hat{\phi}$ is built from entries of $P_\theta(\mathbf{V})$ by addition, multiplication and division, so $\hat{\phi}(P_\theta(\mathbf{V}))$ is a rational function of θ and D is rational.

Suppose $\hat{\phi}$ does not denote Q on Θ. Then D is not identically zero, so writing $D = N/M$ with N, M polynomials, we have $N \not\equiv 0$ on the region where $M \neq 0$. The zero set of a polynomial that is not identically zero has Lebesgue measure zero, so $\{\theta \in \Theta : D(\theta) = 0\}$ has measure zero, as does $\{\theta : M(\theta) = 0\}$, the set on which some conditioning event in $\hat{\phi}$ has probability zero, and the expression is undefined. A draw absolutely continuous with respect to Lebesgue measure on Θ therefore assigns probability zero to both, and with $\tau = 0$ and exact arithmetic the draw refutes with probability one. □

Remark 6 (What part (ii) does not say). Part (ii) is stated with respect to Θ. An expression that denotes Q on Θ while differing from it elsewhere in $\mathcal{M}(G)$ is never refuted, and we do not establish that Θ separates $\mathcal{M}(G)$: that would require showing the fixed latent cardinality is sufficient to witness every disagreement, which we have not done. Part (i) is unconditional.

Proof of Corollary 6. By Theorem 2(ii) the event that one draw fails to refute a $\hat{\phi}$ not denoting Q on Θ has probability zero. The event that all k draws fail to refute is contained in the event that the first draw fails, so it also has probability zero, for every $k \geq 1$. □

Proof of Remark 2. For $\tau > 0$ put $E_\tau = \{\theta \in \Theta : |D(\theta)| \leq \tau\}$. The family is nested decreasing as $\tau \downarrow 0$ with $\bigcap_{\tau > 0} E_\tau = \{D = 0\}$, which has measure zero by the proof of Theorem 2. The draw is a probability measure, hence finite, so continuity from above gives $\Pr(E_\tau) \to 0$ as $\tau \to 0$. □

D.4 METRICS

Proof of Corollary 7. A scored response is wrong exactly when it is a false claim or a miss. With $|\mathcal{I}| = |\mathcal{N}| = |\mathcal{S}|/2$, the number of wrong responses is $f|\mathcal{N}| + m|\mathcal{I}| = (f + m)|\mathcal{S}|/2$. Dividing by $|\mathcal{S}|$ gives $1 - a = (f + m)/2$. Since $m \geq 0$, $f \leq 2(1 - a)$.

For $a \geq 1/2$, both bounds are attained. If all $(1 - a)|\mathcal{S}|$ wrong responses are misses, then $f = 0$ and $m = 2(1 - a) \leq 1$. If all are false claims, then $f = 2(1 - a) \leq 1$ and $m = 0$. □

E Structural Leakage in Semi-Synthetic Construction

This appendix gives the feature-level diagnosis of the leak described in Sec. 3.2, the construction that reduces it, and the leakage that remains.

E.1 Features and classifier

For a bi-directed edge $\{u, v\}$ in an instance with query $P(y \mid \mathrm{do}(x))$, write $\mathrm{pa}(\cdot)$ and $\mathrm{ch}(\cdot)$ for directed parents and children, $d(w) = |\mathrm{pa}(w)| + |\mathrm{ch}(w)|$ for the directed degree of w, and $b(w)$ for its number of bi-directed edges. Table 4 defines the thirteen features. All are permutation-invariant, since a bi-directed edge is an unordered pair; none depends on the variable names.

The classifier is a logistic regression on standardized features. Every reported AUC is held out: for each of the five source networks, the classifier is trained on the other four and scores the held-out network's edges, and the AUC is computed once over the pooled out-of-fold scores. A single-feature AUC instead uses the raw feature value as the score, with no model. A value below 0.5 means the feature predicts real edges; both directions are leakage. We call a feature *balanced* when its single-feature AUC lies in $[0.45, 0.55]$.

Table 4: The thirteen structural features, and their single-feature AUC on our construction (927 edges). Values outside $[0.45, 0.55]$ are in bold.

Feature	Definition	AUC
directed degree sum	$d(u) + d(v)$	**0.574**
directed degree min	$\min\{d(u), d(v)\}$	0.534
directed degree max	$\max\{d(u), d(v)\}$	**0.580**
bi-directed degree sum	$b(u) + b(v)$	**0.442**
bi-directed degree min	$\min\{b(u), b(v)\}$	**0.410**
bi-directed degree max	$\max\{b(u), b(v)\}$	0.480
directed adjacency	1 if a directed edge joins u and v, else 0	0.550
directed distance	shortest directed path, either direction; -1 if none	**0.637**
common parents	$\lvert\mathrm{pa}(u) \cap \mathrm{pa}(v)\rvert$	0.472
common children	$\lvert\mathrm{ch}(u) \cap \mathrm{ch}(v)\rvert$	**0.408**
touches query	1 if u or v is x or y, else 0	**0.562**
vertices	$\lvert\mathbf{V}\rvert$	0.505
bi-directed edges	$\lvert B(G)\rvert$	0.482

E.2 The leak

Consider a construction that hides a latent set in a published DAG, projects onto the remaining variables, and then injects spurious bi-directed edges uniformly at random among unconnected pairs. The label of every edge, real or spurious, is known exactly. On this construction (600 instances, 2,775 edges), the classifier separates real from spurious edges at $\mathrm{AUC} = 0.825$.

Three patterns produce the leak. A latent induces a bi-directed edge between every pair of its observed children, so real edges form cliques and concentrate bi-directed degree. The endpoints of a real edge are siblings under a common latent, so they are often joined by no directed path. Projecting a latent with an observed parent P also adds a directed edge from P to each observed child of the latent, so real pairs share observed parents. Uniform injection reproduces none of these patterns.

E.3 The construction

Two design choices reduce the leak. First, Definition 7 restricts latent sets to root-closed ones, so projection adds no directed edges. Second, spurious edges are injected as the cliques of invented latents, each matched to real edges on a weighted set of ten structural features. On our construction (200 instances, 927 edges), the pooled leakage falls from 0.825 to 0.652, and the construction accepts 14.1% of attempts. Table 5 shows the change for the features that leaked most.

Table 5: AUC under uniform injection and under our construction. Rows are single-feature AUCs; the last row is the held-out pooled classifier.

	uniform injection	our construction
common parents	0.266	0.472
directed degree min	0.321	0.534
bi-directed degree sum	0.299	0.442
bi-directed degree min	0.258	0.410
directed distance	0.585	0.637
pooled classifier	0.825	0.652

E.4 The leakage that remains

Seven of the thirteen features remain unbalanced (Table 4). The largest is directed distance, which the construction makes worse, from 0.585 to 0.637: 37% of real edges join a pair with no directed path, against 14% of spurious ones. The bi-directed-degree features stay below 0.5, because real edges arrive in cliques and concentrate bi-directed degree, while matched spurious pairs are placed one at a time. Removing the remaining separation would require rejecting every instance in which some real pair has no exact structural match. That drops acceptance to 1.5% and leaves graphs too small to be useful, so we use the construction above and report its leakage as a bar a system must exceed (Remark 5).

F Verification of the Labeling Procedure

The evaluation rests on two computations: the labels, computed by ID, and the estimand grade, computed by the verifier. This appendix checks both, reports two defects the verifier found in a published library, and records which model produced each result.

F.1 Canonical known-answer set

We assembled 24 instances whose verdicts are stated in published sources, transcribed independently of any package's test suite. Graph structures were read from source figures rendered at 600–1600 dpi, never from secondary descriptions. The sources are textbook cases encoded from first principles (Pearl, 1995; 2009); Figures 1 and 2 of Shpitser & Pearl (2008), whose captions state the verdicts explicitly; Figures 1(a) and 1(b) of Shpitser & Pearl (2006b), which differ by one directed edge and differ in verdict; and worked examples from Tikka & Karvanen (2017a), transcribed from the printed `graph.formula` definitions.

One instance is flagged. The napkin graph's edge list could not be confirmed against a primary text, so it is recorded in our results as resting on a secondary source and is excluded from the headline count of 24; twenty-five instances are encoded in total. Both implementations return the correct verdict for it.

F.2 Two implementations of ID

Labels are computed by two Python packages developed by different groups: `ananke` 0.5.0, which implements the one-line formulation of ID, and `y0` 0.2.11, which implements ID and its conditional extensions. The packages share no code. Our oracle calls both on every query and halts on any disagreement instead of choosing between them. Across $29{,}537$ random graphs, $19{,}537$ with 4 to 9 vertices and $10{,}000$ with 10 to 14, they disagree on none; across the $\sim 14.9M$ calls made while building and scoring the benchmark, they also disagree on none.

Both packages implement the same published algorithm, so their agreement cannot expose an error in that algorithm or a transcription mistake they share. We therefore also check both against answers taken from the primary papers, never from either package's tests, as described next.

We considered and rejected two other packages. The identification routine of `dowhy` 0.14 takes a directed graph with no way to mark a variable as unobserved, so it cannot represent an ADMG.

The Python port `causal-effect` 0.0.2 crashes on about 30% of instances. We did not test the reference R package `causal-effect`.

F.3 THE VERIFIER

Section 4.3 describes the verifier. This subsection fixes the details needed to reproduce it and to check that Theorem 2 applies.

SCMs. Each bi-directed edge $u \leftrightarrow v$ is materialized as one latent that is a parent of both u and v. Every variable, observed or latent, is binary. For each variable and each configuration of its parents, the conditional distribution is drawn as $w_i = 0.12 + 0.76\,U_i$ for $i = 1, 2$, with $U_i \sim \mathrm{Uniform}(0, 1)$ independent, and then normalized. Each latent's marginal is drawn the same way. The floor of 0.12 keeps every probability away from zero, so no estimand divides by zero. This distribution is absolutely continuous, as Theorem 2(ii) requires, and the guarantee holds relative to binary SCMs (Remark 3).

Exact computation. The joint distribution is computed by enumerating every configuration of the observed and latent variables, and each interventional distribution by removing the edges into the intervened variables. No quantity is estimated by sampling. Main-pool grading uses seeds 101 and 102, one per SCM; the known-answer checks and the case study use seeds 101 to 105.

Grammar. A returned estimand must use four constructs: a probability term $P(\mathbf{a} \mid \mathbf{b})$ over observed variables, a product, a quotient, and a sum over named variables. Subtraction, expectations, numeric constants and other functions are rejected. The grammar covers every expression ID returns, since ID builds its output from exactly these constructs. Every variable must be an observed vertex of the instance, and an estimand may have at most 400 nodes and nesting depth 24. An estimand that fails to parse is counted as a parse failure and is not retried.

F.4 FIVE DEFECTS IN A PUBLISHED CAUSAL-INFERENCE LIBRARY

We applied the verifier to the outputs of `y0`, version 0.2.11, a published causal inference library. On five graphs it returns estimands that are numerically incorrect, while its verdicts remain correct. We reported all five cases upstream.

On the napkin graph the library returns $P(Y \mid X)$ for $P(y \mid \mathrm{do}(x))$, which is precisely the naive conditional the napkin construction exists to refute. That expression deviates from exact SCM ground truth by 7.5×10^{-4}. On Figure 7 of Tikka & Karvanen (2017a), it returns a factorization that deviates from ground truth by 8.6×10^{-3} at the first of our five seeds, with a range of 4.1×10^{-3} to 1.65×10^{-2} across them; the formula published for that graph agrees with ground truth at $\sim 10^{-16}$. Appendix H.3 works this case out in full.

The other three graphs are nine-vertex instances from our main pool. Over five seeds and both intervention values, the library's expressions deviate from ground truth by up to 2.3×10^{-2}, 8.1×10^{-3} and 6.9×10^{-2}. On two of them, the factor for the outcome omits at least one of its parents: for $P(v_6 \mid \mathrm{do}(v_4))$ on one graph, it conditions on V_2 and V_5 and omits the treatment V_4, a parent of V_6. On two of these instances, FLASH and PRO both return correct estimands where the library's is wrong.

All five outputs sum to one, so no discrepancy is a normalization artifact. We confirmed each by hand-coding the returned expression independently of our parser, and the library's own shipped example of the napkin graph reproduces the same output. All five magnitudes come from seeded draws and reproduce from the artifact. The artifact's results file also lists one mismatch for an instance that was never evaluated; it is a bookkeeping entry, not a defect. Our pipeline uses only the library's identifiability verdicts, which are correct in all five cases, so no label is affected. We report the defects because they illustrate the argument before Definition 4: an expression that looks like a plausible estimand, produced by trusted software, can be wrong in a way that only numerical evaluation exposes.

F.5 MODEL IDENTITIES

Identifier	Pin quality	Run dates
`gemini-3.7-flash`	floating alias, requested only	2026-08-17/18
`gemini-3.1-pro-preview`	floating preview alias, requested only	2026-08-17/18
`gpt-5.5-2026-04-23`	dated snapshot, echo-verified	2026-08-20

The three differ in how firmly they are pinned. For `gpt-5.5-2026-04-23` the API returns the served model identifier on every call, and we log the identifier per call. Coverage is 1,189 of 1,189, so the pin is verifiable from the logs and not merely requested. The Gemini client captured no served-model field, so those results rest on the requested alias alone, and one of the two is an explicitly preview endpoint. Neither Gemini result can be re-run against identical weights.

Reasoning. No run set a thinking budget or reasoning effort; every call used the vendor's default, and Gemini calls used temperature 0. Reasoning ran on every call: no logged reply has zero reasoning tokens. Table 6 gives the mean reasoning tokens per reply. The vendor chooses the reasoning depth, so it differs between systems, by about $3\times$ between FLASH and PRO, and is not held fixed in any comparison.

Table 6: Mean reasoning tokens per reply.

	main pool	scale grid
FLASH	2,421	4,114
PRO	8,070	17,144
GPT-5.5	2,994	5,733

The asymmetry deserves naming: the model with the strongest reproducibility guarantee is the one whose results differ most from the other two (Sec. 5.3). A reader inclined to discount the cross-model spread should note that discounting it requires trusting the two endpoints we can *less* well account for. Open-weights models, pinned by full revision hash:

Model	Revision
`Qwen/Qwen2.5-0.5B-Instruct`	`7ae557604adf67be50417f59c2c2f167def9a775`
`Qwen/Qwen2.5-1.5B-Instruct`	`989aa7980e4cf806f80c7fef2b1adb7bc71aa306`
`Qwen/Qwen2.5-3B-Instruct`	`aa8e72537993ba99e69dfaafa59ed015b17504d1`
`sentence-transformers/all-MiniLM-L6-v2`	`1110a243fdf4706b3f48f1d95db1a4f5529b4d41`

G EVALUATION PROTOCOL AND ADDITIONAL RESULTS

Three things determine what a reported figure means: the budget a model was given, the form in which it saw the graph, and the rule by which its reply was turned into a verdict. None is neutral. The budget and the graph format differ across runs, and the parser resolves different models by different routes. This appendix records them. It also gives in full the results the body states in summary.

G.1 PUBLISHED NETWORKS

Table 7 lists the five networks from which the published and semi-synthetic instances are built. All five come from the bnlearn repository (**?**), loaded at a pinned library version. The networks carry no version string, so the loader checks each network's node and edge counts against the repository's published figures and refuses to proceed if they differ.

G.2 OUTPUT BUDGETS

All three models draw internal reasoning from the same budget as visible output, so a budget sized for the answer alone produces truncation or, in one case, a request error with no content at all. An early run at a smaller budget truncated a majority of PRO responses and inverted one of our own reported figures, because the instances dropped were disproportionately those the model reasoned longest about. Budgets were raised and differ by run: 16,384 tokens for the main-pool verdict task;

Table 7: The five networks behind the published family (Definition 7).

Network	Domain	Source	Nodes	Edges
ASIA	lung-disease diagnosis	Lauritzen & Spiegelhalter (1988)	8	8
CHILD	congenital heart disease	Spiegelhalter (1992)	20	25
INSURANCE	car-insurance risk	Binder et al. (1997)	27	52
ALARM	patient monitoring	Beinlich et al. (1989)	37	46
HEPAR2	liver-disorder diagnosis	Onisko (2003)	70	123

32,768 for the scale grid, the second-vendor run, the repeat-sampling study and the justification variant; and per row as given in Table 12 for repair. The oracle condition allows fifteen calls to ID and was run on 150 of the 300 main-pool non-identifiable instances, of which 144 are distinct.

G.3 GRAPH SERIALIZATION

Two renderings of the graph exist in the code base. The main-pool verdict runner emits one edge per line with explanatory parentheticals; every other runner emits a compact form, one line per edge type. The prompt template around the graph block, and the response parser, are identical throughout.

The consequence is confined to one comparison. On the scale grid all three models received prompts identical in bytes: we regenerated the expected prompt for all 600 instances and matched its hash in each model's log, finding 600 of 600 in each and no other prompt hash in any of them. On the main pool, GPT-5.5 received the compact form and the two Gemini endpoints the multi-line form, so the cross-vendor comparison in Table 2 holds the template and the parser fixed while the graph block differs in layout. The designed repeat-sampling study used the compact form throughout.

G.4 HOW A VERDICT IS READ

The parser resolves a response to a verdict by three routes, tried in order. The first is an explicit verdict line, which in every case we observed has the strict format `VERDICT: IDENTIFIABLE` or `VERDICT: NOT IDENTIFIABLE`. The second is a fenced JSON block parsing under the estimand grammar, read as a claim of identifiability. The third is a scan of the whole response for the word *identifiable*. The second route can produce a claim and never a refusal, so it is not symmetric across labels; the third is a fallback we did not document in advance and which fired only on truncated text.

Table 8 gives the decomposition for every model and pool. The reading is straightforward: FLASH and GPT-5.5 are resolved entirely by explicit verdict lines, in both pools and on both labels, and neither fallback fired for either model once. PRO is the only model with mixed routes. This matters most for Sec. 5.3, whose headline comparison is FLASH against GPT-5.5 on the scale grid. All 51 and all 3 of those false claims are explicit verdict lines, as are all 249 and 297 correct refusals behind the denominators. That comparison is therefore like-for-like in route as well as in instance and prompt.

Two further runs were checked the same way. In the justification variant no response from any model contains a JSON block, so neither fallback could fire, and every verdict is an explicit line; FLASH's rise from nine false claims to fourteen is therefore five additional stated claims and not an artifact of the longer replies. In the repeat-sampling study all 600 FLASH samples are verdict lines, and all nine of its non-unanimous instances are changes of stated verdict. PRO has two samples resolved by the estimand route, one of which falls on a non-unanimous instance: on an identifiable instance two samples state `NOT IDENTIFIABLE` and the third returns a bare estimand read as a claim. That is a change of route and not of stated verdict, and Appendix G.10 reports PRO's figure both ways.

G.5 TRUNCATION

FLASH and GPT-5.5 have no truncated responses in either pool: every scale-grid row carries a populated stop reason of `STOP`, and FLASH's longest response is 15,089 tokens of 16,384 on the main pool and 13,973 of 32,768 on the scale grid.

Table 8: How each verdict was read, by model, pool and label. Counts are of responses; the shipped metrics reproduce exactly from them.

		total	verdict line	estimand block	whole-text scan
false claims					
FLASH	main	9	9	0	0
FLASH	scale	51	51	0	0
PRO	main	8	4	2	2
PRO	scale	22	16	6	0
GPT-5.5	main	0	0	0	0
GPT-5.5	scale	3	3	0	0
correct refusals					
FLASH	main	291	291	0	0
FLASH	scale	249	249	0	0
PRO	main	283	281	0	2
PRO	scale	276	276	0	0
GPT-5.5	main	300	300	0	0
GPT-5.5	scale	297	297	0	0

PRO has thirteen main-pool responses cut at the ceiling of $16{,}380$ output-plus-reasoning tokens. Seven yielded no verdict and are among its eleven unscored instances, alongside one formatting failure and three transport errors. Five were resolved by the whole-text scan, two of which are counted as false claims in the shipped metrics and three as correct refusals. One had completed an estimand block before the cut. On the scale grid PRO has three truncated responses, at $n{=}40$ and $n{=}50$. The 0.0000 truncation rate in our shipped scale metrics is a caching artifact: 599 of its 600 rows were served from a log that predates the stop-reason field, and a null value was counted as not truncated.

Table 9 gives the effect of excluding the truncated responses. No comparison in the paper changes, and we report the shipped figures throughout so that every number rests on one treatment.

Table 9: PRO under the shipped treatment and with truncated responses excluded. FLASH and GPT-5.5 are unaffected, having none.

PRO	as shipped	truncated excluded
main-pool accuracy	0.9779 (589 scored)	0.9811 (583 scored)
main-pool false claims	8/291 = 0.0275	6/287 = 0.0209
scale-grid accuracy	0.9415	0.9430
scale-grid false claims	22/298 = 0.0738	21/297 = 0.0707
false claims at $n{=}50$	6/49 = 0.1224	5/48 = 0.1042

G.6 Small open-weights models in full

On edge classification, the small models show no useful signal (Table 10). Across 0.5B–3B models, performance sits at or below chance in every condition, with no trend across the three sizes. The one strongly non-chance value is an inversion: the 1.5B model reaches AUC 0.359 on named instances, as far below chance as 0.641 would be above it, and its scrambled-name control returns to 0.493. The inversion is name-driven and not noise, but it appears in a single cell and we report it as an isolated observation and not a mechanism.

G.7 Scale grid, numerically

G.8 Accuracy and soundness under scale, in full

Across graph sizes from 10 to 50 vertices, the Gemini models lose accuracy slowly while their false-claim rates rise, and GPT-5.5 does neither (Table 11). For PRO, accuracy falls from 96.0% to 90.9%, five points, while the false-claim rate rises from 2.0% to 12.2%, a six-fold increase. FLASH shows the same divergence more steeply. GPT-5.5 holds at 99.5% accuracy with a false-claim rate of 1.0%;

Table 10: Small open-weights models on the edge-classification task, against name-blind and lexical baselines. Values come from the uniform-injection construction of Appendix E; the structural baseline is 0.821 under the original features and 0.825 under the permutation-invariant features of Table 4.Values come from the superseded construction of Appendix E. **Comparisons are valid within a column only**: the published and scrambled columns are scored on the same 2,775 edge instances, the novel column on a separate set of 731, and none of these are comparable to the frontier-model results elsewhere in this paper. The 1.5B model discriminates in the wrong direction; scrambling variable names returns it to chance, so the inversion is name-driven.

Scorer	published	scrambled	novel domains
Random	0.4953	0.4953	0.5342
Embedding similarity	0.5047	0.4115	0.5590
Structural (no names)	0.8210	0.8210	0.7487
0.5B	0.4405	0.4770	0.4529
1.5B	**0.3586**	0.4929	0.5363
3B	0.5181	0.4427	0.5100

Table 11: Verdict accuracy against graph size.

	FLASH		PRO		GPT-5.5	
n	Acc.	False-claim	Acc.	False-claim	Acc.	False-claim
10	0.940	0.12	0.960	0.02	**1.000**	0.00
15	0.960	0.06	0.960	0.04	**1.000**	0.00
20	0.890	0.20	0.960	0.06	0.990	0.02
30	0.840	0.28	0.940	0.10	**1.000**	0.00
40	0.920	0.16	0.919	0.10	**1.000**	0.00
50	0.840	0.20	0.909	0.12	0.980	0.04
overall	0.8983	0.1700	0.9415	0.0738	**0.9950**	**0.0100**

its three errors across six sizes form no trend. Excluding PRO's three truncated responses changes no comparison (Table 9).

With fifty non-identifiable instances per size, the per-size intervals overlap for adjacent sizes, so the shape of the trend within the grid is suggestive, not established. The pooled comparison is resolved: FLASH's false-claim rate is 0.170 $[0.127, 0.217]$ on the scale grid, against 0.030 on the main pool.

The relevant subgraph grows far more slowly than the graph. Identifiability depends only on the subgraph over $\mathrm{An}(y)$, since the first step of ID discards every other vertex (Shpitser & Pearl, 2006b), and by Corollary 4 no minimal repair uses an edge outside it. From $|\mathbf{V}| = 10$ to 50, mean $|\mathrm{An}(y)|$ rises only from 6.9 to 15.1, and the number of bi-directed edges inside $\mathrm{An}(y)$ stays nearly flat, from 4.6 to 5.7. Enlarging the graph therefore adds mostly vertices that are irrelevant to the query, so the degradation is consistent with sensitivity to irrelevant structure more than with a harder identification problem. The spread is wide at every size ($|\mathrm{An}(y)|$ ranges from 4 to 34 at $|\mathbf{V}| = 50$), so size is a coarse proxy for difficulty.

G.9 Repair and enumeration in full

Table 12 reports repair and enumeration on the non-identifiable instances of the main pool, and Table 13 reports FLASH against graph size. The trivial repair $A = B(G)$ is always valid but minimal on only 8.3% of instances, so validity alone is a weak target; the systems' valid repairs are minimal on 96.8–99.3% of instances. With ID as a tool, every repair FLASH returns is valid and minimal, at every size. Exhaustive enumeration of $\mathcal{F}$ becomes infeasible beyond 20 vertices, so frontier coverage and mean $|\mathcal{F}|$ at larger sizes describe only the instances where enumeration finished.

Table 12: Repair and enumeration on the main pool.

Model	Condition	Budget	Parse err.	Validity	Min. | valid	Exact $\mathcal{F}$	Recall
FLASH	one-shot	8,192	0.007	0.9295	0.9675	0.8993	—
FLASH	asked-for-all	32,768	0.003	0.9264	0.9928	0.9197	0.9066
FLASH	oracle	8,192	0.053	1.0000	1.0000	1.0000	—
PRO	one-shot	32,768	0.010	0.9697	0.9826	0.9529	—
PRO	asked-for-all	32,768	0.030	0.9485	0.9746	0.9244	0.8704
PRO	oracle	8,192	0.380	excluded: parse failure on 38% of instances			
trivial ($A = B(G)$)		—	—	1.0000	0.0833	—	—

Table 13: Repair against graph size (FLASH).

n	Valid	Min. | valid	Oracle valid	Oracle min.	$\mathcal{F}$ coverage	mean $\lvert\mathcal{F}\rvert$
10	0.880	0.9545	1.000	1.000	1.00	3.28
15	0.860	0.9535	1.000	1.000	1.00	3.73
20	0.860	0.8837	1.000	1.000	0.88	2.30
30	0.880	0.7500	1.000	1.000	0.10	1.40
40	0.920	0.8696	1.000	1.000	0.02	1.00
50	0.980	0.8163	1.000	1.000	0.00	—

G.10 Reproducibility at temperature zero

Temperature zero did not produce deterministic output in either Gemini model. Drawing three fresh responses for each of 200 instances stratified evenly by label, FLASH returns a non-unanimous verdict on 4.5% of instances (95% CI $[2.0, 7.5]$) and PRO on 2.5% (CI $[0.5, 5.0]$); pairwise flip rates are 3.0% and 1.7%. Two properties of the instability matter more than its size. One of PRO's five non-unanimous instances is a change of parser route and not of stated verdict; setting it aside gives $4/200$ with a pairwise rate of $8/600$ (Appendix G). FLASH's nine are all changes of stated verdict.

It is not symmetric across labels. FLASH is perfectly unanimous on all one hundred identifiable instances and unstable on nine of the hundred non-identifiable ones, so its entire instability sits where a flip converts a correct refusal into a false claim. PRO's flips are spread across both labels, three and two. And it tracks difficulty as accuracy does: FLASH's instability concentrates in the middle density band (1.1% below six bi-directed edges, 8.8% between six and twelve, 5.5% above), the same band in which its verdict errors concentrate.

Majority voting over the three samples does not repair this. Accuracy moves by $+0.8$ points for FLASH and -0.2 for PRO: neither is meaningful at this sample size, and one has the wrong sign, so three times the inference budget recovers nothing. The consequence is methodological and general. A single-run accuracy figure from these endpoints is a sample and not a constant, and a gap of one or two points between two systems, or between a system and itself, lies within the noise of re-running. We report single-sample figures throughout and mark them as such, and we suggest that repeat variance should be standard for benchmarks evaluated against commercial endpoints.

H Worked Examples

The paper claims that two things are computable: whether a query is identifiable, and whether a returned expression is correct. This appendix works one instance through each, then applies both computations to a graph an author published. The first instance is drawn from the pool; the second is a known-answer case from the literature.

H.1 A certified label: a non-identifiable instance and its frontier

Figure 4 shows an instance of the published family drawn from the CHILD network, which models congenital heart disease in newborns. Five observed variables remain after projection: cardiac mix-

ing (CM), duct flow (DF), hypoxia distribution (HD), hypoxia in O_2 (HO) and lower-body O_2 (LB). The query is $P(\mathsf{LB} \mid \mathrm{do}(\mathsf{CM}))$.

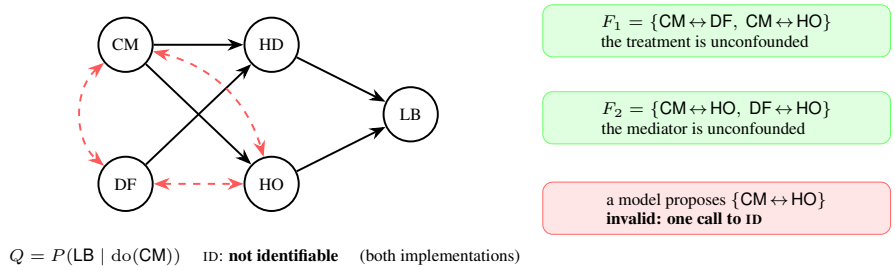


Figure 4: A certified instance, its frontier, and an invalid repair. Dashed edges are unobserved common causes.

The label. Both implementations return FAIL. The obstruction is a hedge (Definition 12): F on $\{\mathsf{CM}, \mathsf{DF}, \mathsf{HO}\}$ and F' on $\{\mathsf{DF}, \mathsf{HO}\}$ are C-forests with the same roots $\{\mathsf{DF}, \mathsf{HO}\}$, both ancestors of LB; F contains the treatment and F' does not. One of the three frontier models named exactly this pair when asked for a justification (Sec. 5.2); all three returned the correct verdict.

The frontier. Exhaustive search over the 2^3 augmentations, seven calls to ID, returns $\mathcal{F}(G, Q) = \{F_1, F_2\}$. No single deletion suffices: each of the three singletons leaves the query non-identifiable. The third pair, $\{\mathsf{CM} \leftrightarrow \mathsf{DF}, \mathsf{DF} \leftrightarrow \mathsf{HO}\}$, also fails.

The two elements have equal cardinality and incompatible meanings. F_1 asserts that nothing unobserved drives both cardiac mixing and either of its neighbors, so the treatment is unconfounded. F_2 asserts that nothing unobserved drives hypoxia in O_2 together with cardiac mixing or duct flow, which makes duct flow a valid adjustment. A procedure ranking repairs by size cannot choose between them, and nothing in the graph can: the choice is a clinical judgment about which confounding is credible. The same pattern recurs in the published graph of Appendix H.4.

An invalid repair. Asked for a minimal restoring set, one model proposed the single edge $\{\mathsf{CM} \leftrightarrow \mathsf{HO}\}$. The model argued that this edge carries the only confounding path, since a collider blocks the remaining path through duct flow. One call to ID shows the proposal invalid (Definition 9), and Theorem 1 shows why: after deleting $\mathsf{CM} \leftrightarrow \mathsf{HO}$, the edges $\mathsf{CM} \leftrightarrow \mathsf{DF}$ and $\mathsf{DF} \leftrightarrow \mathsf{HO}$ still connect F, and F' is untouched, so the hedge survives. The set is not in $\mathcal{A}(G, Q)$, let alone minimal. Two properties of this failure are worth recording. It is a repair error and not a verdict error: the same model called the instance non-identifiable correctly. The graph also occurs twice in the pool, one of eight duplicated keys, and the same wrong repair was returned both times under different reasoning. This is a small instance of the instability reported in Appendix G.10.

H.2 A FALSE CLAIM BY THE STRONGEST MODEL

GPT-5.5 makes three false claims on the scale grid (Table 11). None is truncated or read by a fallback route: each reply opens with `VERDICT: IDENTIFIABLE` and returns a parseable estimand. We work through the one with the smallest hedge.

The instance has 50 vertices, and the query is $P(v_{41} \mid \mathrm{do}(v_{21}))$. ID returns FAIL with a hedge on four vertices: F on $\{V_7, V_{13}, V_{21}, V_{28}\}$ and F' on $\{V_7, V_{13}, V_{28}\}$, with bi-directed edges

$$F^{\leftrightarrow} = \{V_7 \leftrightarrow V_{28},\ V_{13} \leftrightarrow V_{28},\ V_{21} \leftrightarrow V_{28}\}, \qquad F'^{\leftrightarrow} = F^{\leftrightarrow} \setminus \{V_{21} \leftrightarrow V_{28}\}.$$

F contains the treatment V_{21} and F' does not; a single bi-directed edge, $V_{21} \leftrightarrow V_{28}$, joins the treatment to the rest of the hedge. The hedge lies upstream of the outcome: V_{41} is not in F, and the roots of the hedge are its ancestors. The reply states the verdict and returns a chain-rule expansion that conditions each variable on all its predecessors.

Both edge sets are trees, so by Theorem 1 deleting any one of the three edges destroys this hedge (Corollary 3). The other two false claims have the same structure: the hedge contains the treatment

and is joined to it by one or two bi-directed edges. The replies return a truncated factorization in one case and an adjustment for all 32 other ancestors of the outcome in the other. Accuracy counts each of these three answers as one error, the same as a miss, although each asserts an estimand where none exists.

H.3 A VERIFIED GRADE: AN ESTIMAND THAT IS PROPER AND WRONG

The second example shows the grading side. Figure 7 of Tikka & Karvanen (2017a) carries five observed variables with $\mathsf{X} \to \mathsf{Z}_1 \to \mathsf{Y}$, $\mathsf{Z}_2 \to \{\mathsf{X}, \mathsf{Z}_1, \mathsf{Z}_3\}$ and $\mathsf{Z}_3 \to \mathsf{Y}$, with bidirected edges $\mathsf{X} \leftrightarrow \mathsf{Y}$, $\mathsf{X} \leftrightarrow \mathsf{Z}_2$, $\mathsf{X} \leftrightarrow \mathsf{Z}_3$ and $\mathsf{Y} \leftrightarrow \mathsf{Z}_2$. The query is $P(z_1, z_2, z_3, y \mid \mathrm{do}(x))$, which Tian & Pearl (2002) proved identifiable; both our implementations agree.

Two expressions are in play. The published estimand is

$$P_x(z_1, z_2, z_3, y) \;=\; P(z_1 \mid x, z_2) \, \textstyle\sum_{x'} P(y, z_3 \mid x', z_1, z_2) \, P(x', z_2),$$

and `y0` 0.2.11 (Appendix F.4) returns instead

$$\widehat{\phi} \;=\; P(z_1 \mid x, z_2) \, P(z_3 \mid z_2) \, P(y \mid z_2, z_3) \, P(z_2).$$

Both are proper distributions: each sums to one over the sixteen configurations, to within 10^{-12}. Nothing about their surface form marks either as wrong, and a grader comparing strings would have to decide which is the reference.

The evaluation. We draw a binary discrete SCM realizing the graph, with one binary latent per bi-directed edge, and enumerate the joint exactly. Table 14 gives every cell at $\mathrm{do}(x{=}1)$ for the first of our five seeds. The published estimand agrees with ground truth to 5.6×10^{-17} in every cell. The library's expression deviates by up to 8.6×10^{-3}, and across the five seeds the maximum deviation ranges over 4.1×10^{-3} to 1.65×10^{-2}.

Table 14: $P(z_1, z_2, z_3, y \mid \mathrm{do}(x{=}1))$ under one seeded SCM. Ground truth by exact enumeration; the published estimand matches it to 5.6×10^{-17} throughout and is omitted. Deviations cancel within each (z_1, z_2, z_3) block.

z_1	z_2	z_3	y	truth	library	difference
0	0	0	0	0.068800	0.066377	−0.002423
0	0	0	1	0.087453	0.089876	+0.002423
0	0	1	0	0.073608	0.066449	−0.007159
0	0	1	1	0.083090	0.090249	+0.007159
0	1	0	0	0.028730	0.028270	−0.000460
0	1	0	1	0.039616	0.040076	+0.000460
0	1	1	0	0.050490	0.045796	−0.004694
0	1	1	1	0.059443	0.064138	+0.004694
1	0	0	0	0.063194	0.065219	+0.002025
1	0	0	1	0.090332	0.088308	−0.002025
1	0	1	0	0.056682	0.065289	+0.008608
1	0	1	1	0.097282	0.088674	−0.008608
1	1	0	0	0.031453	0.031917	+0.000464
1	1	0	1	0.045710	0.045246	−0.000464
1	1	1	0	0.046903	0.051704	+0.004801
1	1	1	1	0.077213	0.072412	−0.004801

Why this example matters more than its magnitude. The deviations in each (z_1, z_2, z_3) block are equal and opposite, so summing Y out of the library's expression recovers ground truth to 8×10^{-17}. Its answer is therefore correct for $P_x(z_1, z_2, z_3)$ and wrong only for the joint that was asked for. An evaluation checking a marginal, or a single scalar contrast, would see nothing. The error is confined to the factor for Y, which the returned expression writes as $P(y \mid z_2, z_3)$. That factor conditions on neither z_1, a directed parent of Y, nor on x. We record this as an observation about the expression and claim no diagnosis of the implementation.

This is what Theorem 2 buys in practice. The library's verdict is correct, its expression is a proper distribution, its marginal is exact, and it is still not the requested quantity. Only evaluation against an SCM whose interventional distribution is known separates the two.

Provenance. The graph is transcribed from the printed `graph.formula` definition in the source, not from a secondary description. Parameters for the seeded model, the full-precision values, and the same table at $\mathrm{do}(x{=}0)$ are in the artifact.

A refuted estimand from a model. For $P(v_4 \mid \mathrm{do}(v_0))$ on a five-vertex random instance, PRO returned

$$\hat{\phi} = \sum_{v_2,v_3} P(v_2 \mid v_0, v_3)\, P(v_3)\, P(v_4 \mid v_0, v_2, v_3),$$

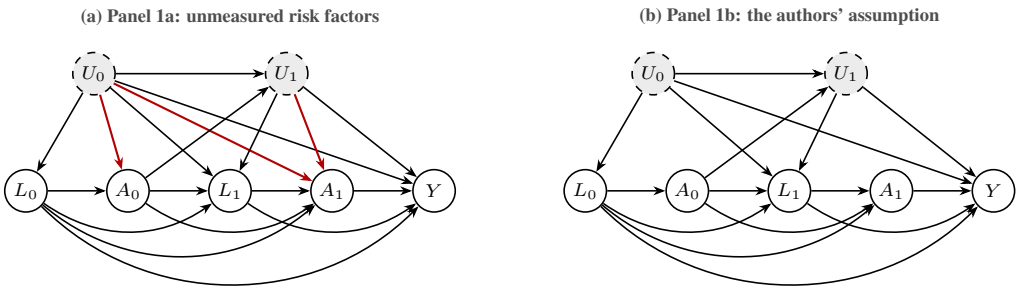
while ID returns

$$\phi = \sum_{v_2,v_3} P(v_2 \mid v_0, v_3) \sum_{v_0'} P(v_0' \mid v_3)\, P(v_3)\, P(v_4 \mid v_0', v_2, v_3).$$

The two differ by one summation: ϕ averages the last factor over the observed distribution of the treatment, while $\hat{\phi}$ evaluates it at the intervened value. The verifier refutes $\hat{\phi}$ with a deviation of 1.2×10^{-3}. All four refuted model estimands share this error, a missing summation over the treatment.

H.4 A PUBLISHED GRAPH: ROBINS, HERNÁN AND BRUMBACK (2000)

The instances above are generated. To show the same machinery applied to a graph an author drew and published, we take Figure 1 of Robins et al. (2000), the methods paper introducing marginal structural models, whose companion study estimates the effect of zidovudine on survival in HIV-positive men (Ryalen et al., 2026). We chose it because its authors mark latent variables explicitly: the prose defines U_k as all unmeasured causal risk factors for the outcome, and states that panel 1b differs from 1a only in that the arrows from those risk factors into the treatment variables are removed. Latent variables appear in a substantial minority of applied graphs, roughly 37% of the applied health DAGs reviewed by Tennant et al. (2021), so this is a real class of instance and not a curiosity. We transcribed the figure from the scanned original; two edges we could not settle were run as four variants, and every verdict and every frontier below is identical across them. Figure 5 shows our encoding.



(c) Two minimal restoring sets for $P(y \mid \mathrm{do}(a_1))$ in (a), each of size four

$\{A_0 \leftrightarrow A_1,\ A_1 \leftrightarrow L_0,\ A_1 \leftrightarrow L_1,\ A_1 \leftrightarrow Y\}$
the treatment A_1 is unconfounded

$\{A_0 \leftrightarrow Y,\ A_1 \leftrightarrow Y,\ L_0 \leftrightarrow Y,\ L_1 \leftrightarrow Y\}$
the outcome Y is unconfounded

Figure 5: Our encoding of Figure 1 of Robins et al. (2000). A_t is zidovudine treatment at time t, L_t the measured risk factors, Y the outcome, and U_t the unmeasured risk factors (dashed, latent). **(a)** Projecting out U_0 and U_1 confounds every pair of observed variables, so the joint effect and both single effects are non-identifiable. **(b)** The authors' assumption removes the three red edges, from unmeasured risk factors into treatments; only L_0, L_1 and Y stay confounded, and all three effects become identifiable. **(c)** For the effect of A_1 in (a), two minimal restoring sets have the same size and opposite meaning, so cardinality cannot choose between them. Two edges we could not read from the scan, $A_0 \to Y$ and $L_0 \to U_1$, change neither projection and are omitted.

The authors' claims check out. Both implementations agree on all six queries: the joint effect and both single effects are non-identifiable in panel 1a and identifiable in panel 1b. On 1b the authors' g-formula matches exact SCM ground truth at 2.2×10^{-16}, while the naive estimand they warn against is wrong by 7.0×10^{-3} on the same models. Their central methodological point is confirmed numerically on their own graph.

Their identifying assumption is sufficient but not minimal. The 1a-to-1b move deletes seven bidirected edges, every one incident to a treatment. For the joint effect the frontier has five elements, and one is the authors' set with $A_0 \leftrightarrow A_1$ restored: an unmeasured common cause of the two treatment decisions is harmless once both are intervened on. Every route deletes the two treatment–outcome edges, which is the necessary core. For the single effect $\mathrm{do}(a_1)$ the frontier has eight elements, two of which have the same cardinality and opposite meaning: one makes the treatment A_1 unconfounded, the other the outcome Y (Figure 5c). Cardinality cannot separate them, and domain judgment must.

One semantic caveat belongs with this. Frontier elements are defined over the ADMG, and a single latent in the authors' DAG can induce several bi-directed edges; deleting one and keeping another therefore posits a different latent structure and does not erase one of their arrows. This is correct for $\mathcal{A}(G, Q)$ as defined in Sec. 3.3 and worth stating before a frontier element is read back as advice.

I Discussion

I.1 What the results do and do not say about causal competence

Deciding identifiability on a random ADMG with neutral variable names draws on no memorized association between variables and no domain knowledge attached to their names, because there are none to draw on. The accuracy we observe on that family is therefore difficult to square with accounts of language-model causal ability as retrieval (Chi et al., 2024; Zečević et al., 2023) or as knowledge extraction from metadata (Kiciman et al., 2023). Those accounts were established on an earlier model generation, and the most we claim is that they warrant re-examination against current models on tasks where the retrieval route is closed by construction.

The bound in the other direction is firmer. Identification given a graph is a formal task on which a sound and complete polynomial algorithm happens to exist. That is unusual. Competence here implies nothing about causal reasoning where no such procedure is available: discovering the graph, choosing what to measure, deciding whether an assumption is credible. The small open-weights models of Appendix G.6 were tested only on edge classification, so this paper says nothing about where the ability appears as models grow. A reader who takes from this paper that language models can do causal inference has taken more than the measurement supports.

I.2 Using these results in practice

The practical consequences run in two directions, and the more important one is not about evaluation at all. For anyone with a graph and a query, the results argue for calling ID. The algorithm is polynomial, sound and complete, and freely implemented. It decides in milliseconds a question on which the best model we tested still asserts an answer to one in a hundred unanswerable queries. Where a language model helps is in the surrounding work: reading a graph out of a paper, proposing which confounders to consider, explaining why a query fails. With ID as a tool, a model's repairs are valid and minimal at every size we tested, including sizes where our exhaustive enumeration is infeasible (Sec. 5.4). The configuration to avoid is the one our main results measure: a model deciding identifiability unaided and reporting a confident estimand.

For anyone evaluating such a system, three things follow. Report the false-claim rate separately from accuracy, which does not determine it (Corollary 7). On our instances the false-claim rate spans seventeen-fold while accuracy spans ten points, so a ranking by accuracy presents the models as close competitors. Measure on your own distribution of graphs, because degradation with graph size appeared in one model family and not another, so it cannot be inferred from published numbers. And re-run before believing a small gap. At temperature zero, both endpoints we re-sampled return non-unanimous verdicts on a few percent of instances, which exceeds the accuracy gaps between our systems on the main pool (Table 2).

When a query is not identifiable, the frontier names the assumption sets that would restore it, and both the published graph of Appendix H.4 and the worked example of Appendix H.1 show the same thing about how that output should be read. Frontier elements of equal size can carry incompatible meanings, so cardinality does not rank them and no graphical computation can. The frontier narrows

the question to a short list of substantive assumptions; choosing among them is domain work and remains so.

I.3 Applications of this work

Language model research. Two questions are open in language model research. The first is whether a model can tell that a question has no answer. The second is whether a model's stated reasoning reflects how the model reached its verdict. Work on model self-knowledge and abstention studies the first (Kadavath et al., 2022; Kirichenko et al., 2026), and the chain-of-thought faithfulness literature studies the second (Turpin et al., 2023; Lanham et al., 2023; Chen et al., 2025). Both lack certified ground truth. In abstention work, whether a question is answerable is fixed by human judgment, so a model that declines may be right while its label is wrong. In faithfulness work, the correct reasoning is unknown, so a stated justification can be checked for consistency but not for truth. CertID removes that ambiguity for one well-defined domain. Unanswerability here is a theorem, instances can be regenerated after any training cutoff, and the false-claim rate isolates the error an abstention study most needs to measure. The warrant results of Sec. 5.2 give faithfulness research a second tool: because the obstruction that blocks identification is known exactly, a model's stated justification can be checked against the true one.

Cognitive and behavioral modeling of language models. A growing line of work studies language models as subjects of psychological experiment, using paradigms from cognitive science to characterize behavior beyond aggregate accuracy (Binz & Schulz, 2023; Hagendorff et al., 2023; Coda-Forno et al., 2024). Three levels can be distinguished in such work: what a model does, what the model reports about its reasoning, and the internal state that produces the behavior. Existing tasks typically fix the correct response but not the correct reasoning, so self-report can be compared with behavior but not with the truth. CertID fixes both. The certified label is ground truth for the verdict, and the graphical obstruction is ground truth for the justification. Sec. 5.2 reports a dissociation between hidden reasoning and self-report: models reason internally at length yet emit a bare verdict, and name the correct obstruction only when asked. The task also bears on whether a model forms an internal model of the structure it reasons about. Random graphs with neutral names offer no memorized association, so correct verdicts there are behavioral evidence that the graph itself is represented. Probing internal activations against certified labels could test that hypothesis directly.

Human–AI interaction and reliance. In a conversation with a language model, a user sees only what the model emits. Reliance on that output is appropriate when the person accepts correct advice and rejects incorrect advice (Schemmer et al., 2023). The older distinction between misuse and disuse of automation points toward the same target (Parasuraman & Riley, 1997). Over-reliance, the acceptance of incorrect advice, is common and resists simple remedies such as explanations (Buçinca et al., 2021). A false claim on an unanswerable causal query is such a failure: high accuracy conceals the error, and no observational data can reveal one. Our results sharpen the problem for longitudinal use, in which the same person returns to a system over time and the human–AI relationship develops through repeated interaction. First, a correct verdict arrives by default with no reason, so the user has nothing to check against. Second, trust is ordinarily calibrated from observed outcomes, and an error that no observational data can expose produces none. Repeated interaction therefore cannot teach the user when to distrust the system on exactly these queries. Third, verdicts vary across repeated requests at temperature zero (Appendix G.10), so one question may receive different answers in successive sessions. Certified labels make each of these failures observable, so appropriate reliance can be measured where reliance matters most.

Training with verifiable rewards. Reinforcement learning with verifiable rewards trains a language model against a deterministic check of its output, and the approach underlies recent reasoning models (Lambert et al., 2024; Shao et al., 2024; Guo et al., 2025). A deterministic check is harder to exploit than a learned reward model, which invites reward hacking (Skalse et al., 2022). The approach is confined, however, to domains where such a check exists, which in practice means mathematics and code. Identification, including the decision that a query has no answer, has had no such check, and CertID supplies one. The certified label verifies the verdict, and the numerical evaluation verifies the estimand. Instances can be generated without limit and after any training cutoff, so a held-out set stays uncontaminated. Two features of the setting bear on reward design. First, a correct estimand admits many equivalent forms, so a reward based on answer matching would penalize correct outputs; numerical grading does not. Second, the two errors are not equally costly, and

a reward can weight a claimed answer to an unanswerable query more heavily than a refusal of an answerable one. One caveat applies. Theorem 2 guarantees refutation only on the family of binary SCMs the verifier samples (Remark 6). A policy optimized against the verifier could therefore learn expressions that agree on that family and differ elsewhere. Widening the family and the compared intervention values narrows that gap.

I.4 LIMITATIONS

The scope limits are structural and not incidental. Every instance supplies the graph, so nothing here measures whether a model can recover structure from data or from text. That step is where most applied work spends its effort, and where errors are likeliest. Identifiability is settled before any data exist, so a correct verdict says nothing about estimation, finite-sample behavior, positivity, or whether the modeling assumptions hold. The labels certify what is recoverable in principle and are silent on whether the graph is right: a certified verdict on a wrong graph is a certified answer to the wrong question. Finally, the grading compares expressions numerically at a single intervention value over two draws (Sec. 4.3, Appendix C.2), so it certifies the slice compared and not the whole interventional distribution. A good score on CERTID is therefore evidence that a system decides a formal question correctly, and nothing more.

The empirical claims have narrower limits. We evaluate three frontier models from two vendors, each at its vendor's default reasoning depth, which differs between them; the two Gemini endpoints are floating aliases and cannot be re-run against identical weights. Estimands are graded only on graphs of at most 9 vertices, which leaves 41–42% of correct claims unchecked, and GPT-5.5's estimands are not graded. Every figure is a single sample, and repeated sampling at temperature zero returns non-unanimous verdicts on 2.5–4.5% of instances (Appendix G.10). Theorem 2 holds relative to binary SCMs (Remark 6).