ALICE: AMORTIZED LEARNING FOR INVARIANCE-AWARE CAUSAL ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Causal Discovery (CD) is fundamental across domains such as biology, finance, weather, and physical systems. Classical CD approaches often rely on restrictive assumptions about causal structure, noise distributions, and mechanisms, and require sufficiently large datasets for reliable estimation. These factors can limit their effectiveness on real-world datasets. Moreover, classical and existing zero-shot approaches do not explicitly exploit feature semantics, which can provide complementary causal information. To this end, we first introduce **Amortized Learning for Invariance-aware Causal Estimation (ALICE)**, a causal discovery foundation model, trained entirely on synthetic datasets generated from diverse structural causal models, that maps observational data to a full directed causal graph in a single forward pass. Our diverse pretraining corpus spanning graph families, mechanisms, and noise processes enables effective zero-shot transfer to real-world datasets. Second, we propose targeted augmentations and consistency losses to enforce key graph invariances, including realization, permutation, and scale. Leveraging these invariances, we propose test-time optimization strategies that improve the F1 score by up to 14%. Finally, we incorporate feature semantics as a post-processing signal, yielding up to 18% improvement on real-world causal estimation tasks. Across synthetic, biological, physical-system, bivariate, and tabular benchmarks, ALICE achieves accurate and robust causal graph estimation without dataset-specific training.

1 INTRODUCTION

Inferring causal relationships (Zhang et al., 2023) from observational and interventional data has significant impacts in multiple domains, including finance (Sadeghi et al., 2023), microbiology, and weather (Runge et al., 2019). Interestingly, in the real world, the available datasets for causal discovery are structured in the form of tables with hundreds of variables/columns or multivariate time series, where variables share lagged relationships. This simplifies the problem of causal discovery to identifying a directed acyclic graph (DAG) that represents the structured observational datasets.

The classical causal discovery (CD) literature contains multiple dataset- or domain-specific solutions (Spirtes et al., 2001; Chickering, 2002; Hauser and Bühlmann, 2012; Zheng et al., 2018; Shimizu et al., 2006) with strong assumptions about the functional forms of the underlying causal models, noise distributions, etc. When the observational data adhere to these assumptions, these dataset-specific approaches work well. However, real-world datasets typically violate these assumptions.

Following the recent success of tabular (Wang et al., 2026; Hollmann et al., 2025; QU et al., 2025) and multivariate time series foundation models (Ansari et al., 2025; Cohen et al., 2026) trained on large-scale synthetic datasets, recent approaches (CDFM (Qiao et al., 2026) and TabCausal (Li et al., 2026b)) have focused on amortized causal discovery by training on millions of synthetic tables or multivariate time series datasets with the objective of regressing the corresponding DAG. Since synthetic datasets can be devoid of any functional and noise distribution assumptions, these approaches learn to extract DAGs for datasets where classical approaches fail, and generalize to real-world datasets. However, they fail to adhere to key graph invariances pertaining to mechanisms, permutation, and realization.

Overall, we identify three key challenges for an ideal causal discovery model. ❶ **Assumption-free estimation:** The model must not rely on assumptions such as linearity, additive non-linear effects,

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

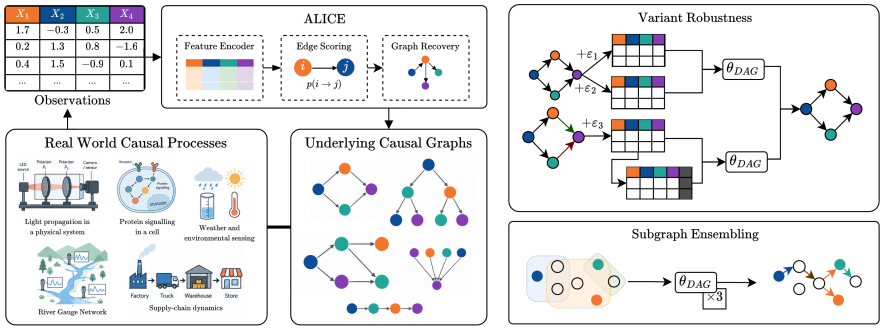


Figure 1: **Amortized Learning for Invariance-aware Causal Estimation (ALICE).** **(Left)** ALICE is trained on millions of synthetic tables generated from known structural causal models (SCMs) with diverse causal and structural relationships which allow for seamless real-world generalization. **(Right, Top)** Our proposed method is robust to different realizations of the same SCM, altered edge mechanisms (differently colored arrows) and independent features (gray column). **(Right, Bottom)** We propose ensembling techniques to improve the model performance at test-time.

Gaussian noise distribution, etc. Rather, an ideal approach should recover the causal structure under arbitrary, unknown mechanisms. ❷ **Preserving invariances:** The model should remain insensitive to input perturbations and transformations that leave the causal process unchanged. Different realizations of the same generative process, reordering and transformations of features should not change the model's predictions. ❸ **Exploiting feature semantics:** The physical meaning of features carries orthogonal information about causal signals that purely numerical methods ignore. The model should be able to seamlessly integrate this orthogonal information and fuse it with the observed values. In this work, we address these challenges through the following core contributions:

- We introduce **Amortized Learning for Invariance-aware Causal Estimation** (**ALICE**, see Figure 1), an attention-based architecture for amortized causal discovery that jointly reasons over observations and features and directly scores every ordered feature pair. ALICE maps an observational dataset into an adjacency matrix zero-shot, and scales to a varying number of observations and features.
- We propose targeted augmentations and pretraining objectives that **enforce invariance by construction**. Specifically, we enforce the fundamental invariances of causal discovery, such as realization, permutation, width, mechanism, and scale.
- Exploiting the invariance-aware predictions, we propose **test-time optimization algorithms** that improve the DAG estimation over the base model's performance. In parallel, we incorporate feature semantics into ALICE's predictions during test time to improve DAG estimation.
- ALICE is trained on millions of diverse synthetic tables to enable effective real-world generalization. It achieves state-of-the-art accuracy, **outperforming frontier CD approaches** across tabular, biological, and bivariate benchmarks, while adhering to key graph invariances.

2 Background and Related Work

First, we explain the basic notation used in the rest of the manuscript.

Structural Causal Models. Consider a system of N_{feat} features $\mathbf{X} = (X_1, X_2, \ldots, X_{N_{\text{feat}}})$, whose causal relationships are represented by a directed acyclic graph (DAG) $G = (V, E)$, with $V = \{1, \ldots, N_{\text{feat}}\}$. We denote its adjacency matrix by $A \in \mathcal{A}_{N_{\text{feat}}} \subseteq \{0,1\}^{N_{\text{feat}} \times N_{\text{feat}}}$, where $A_{ij} = 1$ indicates a directed causal edge $X_i \to X_j$, and $\mathcal{A}_{N_{\text{feat}}}$ denotes the set of adjacency matrices corresponding to DAGs over N_{feat} features. In particular, $\mathcal{A}_{N_{\text{feat}}}$ is a set of matrices without directed cycles, and $A_{ii} = 0$. Overall, let $\mathcal{A} = \bigcup_{n \geq 1} \mathcal{A}_n$. Although graph size is not fixed, $\mathcal{A}$ is countable. Therefore, we define P_{graph} as a discrete distribution over $\mathcal{A}$ or all variable-sized graphs.

We assume that the N_{feat} features are generated by a structural causal model (SCM, Bongers et al. (2021)). For a given graph $A \in \mathcal{A}_{N_{\text{feat}}}$, we define the SCM with N_{feat} features as $\mathcal{M} = (\mathbf{g}, P_\epsilon)$, where $\mathbf{g} = \{g_1, \ldots, g_{N_{\text{feat}}}\}$ denotes the collection of functions, and $\epsilon = (\epsilon_1, \ldots, \epsilon_{N_{\text{feat}}})$ is the list

of exogenous noise variables sampled from the joint noise distribution for all features, P_ϵ. Each feature is generated according to $X_i = g_i\big(X_{\mathbf{pa}_A(i)}, \epsilon_i\big)$, where $\mathbf{pa}_A(i) = \{j : A_{ji} = 1\}$ denotes the set of direct causal parents of X_i. More generally, for a given causal graph (A), we sample $\mathcal{M}$ from a collection of parametric functions and noise distributions, along with their coefficients, denoted by $\mathcal{M} \sim P_{\text{SCM}}(\cdot|A)$. The SCM $\mathcal{M}$ induces an observational distribution $P_{\mathcal{M}}(\mathbf{X})$. An observational dataset consists of N_{obs} independent and identically distributed (i.i.d.) realizations from this distribution, $D = \{\mathbf{x}^{(n)}\}_{n=1}^{N_{\text{obs}}}$, which we equivalently represent as a dataset $D \in \mathbf{R}^{N_{\text{obs}} \times N_{\text{feat}}}$.

Statistical and Classical Causal Discovery. Classical CD recovers the structure of the graph from conditional independence tests or from score optimization under assumptions about the generating mechanism. Constraint-based methods, such as the Peter-Clark (PC) algorithm (Spirtes et al., 2001) estimate the DAG up to a Markov equivalence class. Similarly, score-based methods such as GES (Chickering, 2002) employ greedy search-based techniques over the equivalence class to update graph edges, while NOTEARS (Zheng et al., 2018; 2020) and more recent works (Bello et al., 2022; Montagna et al., 2023; Sethuraman et al., 2023) frame CD as an optimization problem with assumptions about relationships. Alternatively, functional approaches (Shimizu et al., 2006; Hoyer et al., 2008) make structural assumptions about the node relationships, and use non-linearity and non-Gaussianity for identifiability. For time series, Granger Causality (Granger, 1969) tests the existence of lagged cause-effect relationships. **Our key observation** is that statistical methods, while based on first principles, fail to generalize to real-world datasets when the assumptions are violated.

Amortized Zero-Shot Causal Discovery. Causal Discovery Foundation Models, such as TabCausal (Li et al., 2026b), CDFM (Qiao et al., 2026), and FoundCause (Blöbaum et al., 2026) among others (Thompson et al., 2026; Peng et al., 2026) train on large-scale pairs of synthetic observational datasets and their corresponding DAGs (Amortized) to perform zero-shot CD on real-world datasets (**sim-to-real transfer**). This line of work is inspired by the recent success in developing foundation models for tabular (Hollmann et al., 2025; Wang et al., 2026; QU et al., 2025) and time series (Ansari et al., 2025; Cohen et al., 2026), where these models provide accurate zero-shot predictions on real data by training on a diverse collection of synthetic, structured datasets. However, **our key insight** is that these zero-shot CD approaches do not fully exploit the invariance properties of the underlying causal graph. We demonstrate that by incorporating these invariance properties into zero-shot CD approaches, we can significantly improve zero-shot generalization to real-world datasets.

Problem Description. Let P_{graph} denote a distribution over all variable-sized graphs $\mathcal{A}$ and let $P_{\text{SCM}}(\cdot|A)$ denote the SCM distribution over diverse parametric functions and noise distributions, conditioned on $A \in \mathcal{A}$. For each sampled SCM $\mathcal{M}$, an observational dataset $D \sim P_{\mathcal{M}}$ of N_{obs} i.i.d. observations with N_{feat} features can be generated. Our objective is to train a parameterized causal discovery model $\theta_{\text{DAG}} : D \to \hat{A}$, such that

$$\theta_{\text{DAG}} = \arg\min_{\theta} \; \mathbb{E}_{A \sim P_{\text{graph}}} \mathbb{E}_{\mathcal{M} \sim P_{\text{SCM}}(\cdot|A)} \; \mathbb{E}_{D \sim P_{\mathcal{M}}} \left[\ell \left(\theta(D), A \right) \right], \tag{1}$$

where ℓ computes the mismatch between the predicted and ground-truth causal structures. At inference time, we are given only an observational dataset $D^\star \sim P_{\mathcal{M}^\star}$ generated by an unseen SCM $\mathcal{M}^\star$, and we aim to estimate its true causal graph $A^\star = \theta_{\text{DAG}}(D^\star)$ zero-shot.

3 Amortized Learning for Invariance-aware Causal Estimation

In this section, we will describe our approach, **Amortized Learning for Invariance-aware Causal Estimation (ALICE)** for zero-shot causal discovery. One of the key challenges in developing such a model is the absence of large-scale paired combinations of observational data and the corresponding underlying DAG. Therefore, we focus on training on large-scale synthetic observational datasets generated from randomly initialized DAGs.

3.1 Large-Scale Synthetic Training Corpus

The primary objective for a large-scale synthetic training corpus is to cover a wide range of SCMs or DAG structures, such that the trained model can efficiently transfer to real-world scenarios without

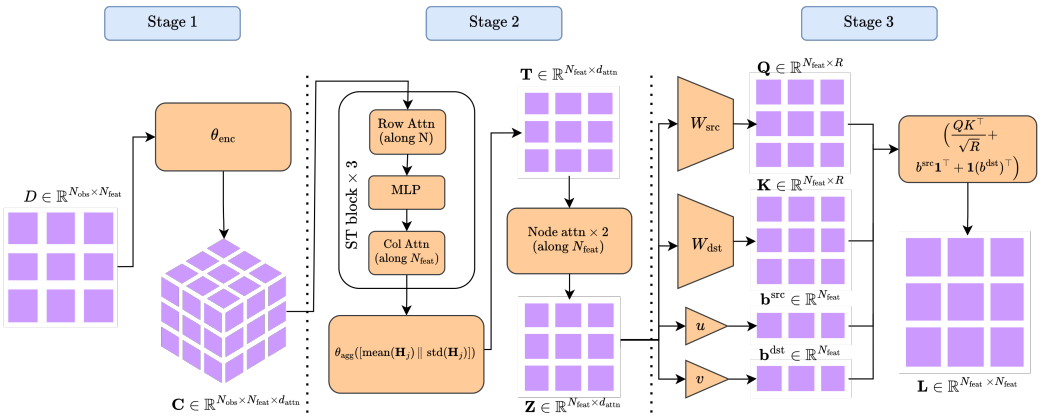


Figure 2: **ALICE architecture.** ALICE effectively converts a dataset D with N_{obs} observations and N_{feat} features into an adjacency matrix A. In stage 1, every cell is projected into a token using the feature and target encoders. In stage 2, through alternating row-column attention we learn the interaction between features and observations. We perform set aggregation to obtain one embedding per feature, and we also allow the feature embeddings to attend each other. Finally, in stage 3, through asymmetric dot products, we identify the presence and the direction of an edge between all pairs of features, represented as the adjacency matrix. See Appendix D for the full details.

any training. We introduce the diversity in SCMs along four axes: ❶ DAG structures, ❷ mechanisms or functional relationships that connect nodes together, ❸ noise (distribution, heteroscedasticity and additive vs. multiplicative entries), and ❹ sampling regime (row-exchangeable cross-sectional vs. temporally ordered).

In **Axis 1**, we sample DAGs with diverse structures (Table 3), such as chains, stars, and trees. Typically, we generate graphs with 2 to 100 features with edge densities in the range of 0.1 to 0.2. In **Axis 2**, mechanisms are sampled homogeneously or independently per parent-child edge from MLPs, regression trees, piecewise-linear, polynomial functions, etc. A detailed list of mechanisms is provided in Table 5. In **Axis 3**, exogenous noise is sampled from various distributions, such as Gaussian, Laplace, Student-t, and Cauchy, with the noise being injected using additive or multiplicative approaches. In **Axis 4**, we sample either i.i.d. cross-sectional datasets with 128 to 1,024 observations or temporally ordered series with distributed-lag causal effects. Overall, the synthetic dataset generation procedure can then be written as

$$A \sim P_{\text{graph}}, \qquad \mathcal{M} \sim P_{\text{SCM}}(\cdot \mid A), \qquad D \sim P_{\mathcal{M}}. \tag{2}$$

We denote the resulting joint distribution over (D, A) by P_{train}. We provide a detailed analysis of our synthetic datasets in Appendix C. The notation $P_{\text{SCM}}(\cdot \mid A)$ does not impose a restriction on the sampled parameterized functions or noise distributions for a particular graph type. It primarily indicates that conditioning on A specifies the parent inputs to each structural equation $X_i = g_i\big(X_{\mathbf{pa}_A(i)}, \epsilon_i\big)$, while allowing diverse mechanism families, parameter values, and noise distributions.

3.2 ALICE Architecture and Synthetic Pre-training

ALICE is an attention-based architecture (see Figure 2) that effectively condenses N_{obs} observations in an observational dataset $D = \{\mathbf{x}^{(n)}\}_{n=1}^{N_{\text{obs}}}$ into one embedding per feature, and then predicts the probability of every ordered pair of feature embeddings being a directed edge. A shared **feature encoder** maps the per-feature normalized values to d_{attn}-dimensional embeddings. Our training randomly alternates between *target-aware* mode, which adds learned embeddings of a randomly selected target, using a target encoder that is different from the feature encoder, to all feature tokens within each observation, and *null* mode, which appends a learned null feature, indicating the absence of a privileged target column.

Consequently, a few spatio-temporal attention blocks capture interactions across observations and features. Then, a **set aggregator** pools the mean and standard deviation of each feature's representations across observations. Further feature-wise attention produces the graph-aware embeddings, and

	Dataset	Metric	ALICE (Ours)	FoundCause	CDFM	TabCausal	CauScale	AVICI	PC	DirectLiNGAM
Real-World	Causal Chambers	AUROC	**0.9564**	0.7521	0.9353	0.9386	0.7023	0.2860	0.5333	0.7584
		F1	0.7213	0.6765	**0.7273**	0.5970	0.4324	0.0000	0.1538	0.2041
		SHD	**17**	22	**17**	27	40	41	47	37
	Tuebingen	Weighted Accuracy	**0.6289**	0.6251	0.5605	0.6168	0.2292	0.5243	0.5000	0.5280
	Causal Rivers	AUROC	**0.7447**	0.7146	0.5843	0.6623	0.5882	0.5870	0.5210	0.6854
		F1	**0.4980**	0.0604	0.3951	0.3137	0.2246	0.2942	0.0664	0.3896
	Sachs	AUROC	**0.7476**	0.7337	0.6818	0.6486	0.5832	0.5984	0.5614	0.6135
		F1	0.3077	0.2963	**0.3333**	0.3200	0.1667	0.2500	0.2400	0.2000
		SHD	**13**	14	17	14	15	15	14	15
Synthetic	Synthetic Suite	AUROC	**0.9317**	0.9187	0.8485	0.8333	0.8153	0.7724	0.6594	0.7089
		F1	**0.6148**	0.5566	0.4601	0.5234	0.3360	0.3467	0.3832	0.2062
	Causal Arena Synthetic	AUROC	0.8658	**0.8968**	0.8264	0.8172	0.7968	0.7341	0.6727	0.6971
		F1	**0.5497**	0.5429	0.4224	0.4647	0.3688	0.3221	0.4010	0.2148
	Causal Arena Formula	AUROC	0.8481	**0.8661**	0.8216	0.8274	0.6816	0.7563	0.6076	0.6568
		F1	**0.4347**	0.4204	0.4058	0.4111	0.2149	0.3156	0.2821	0.2520

Table 1: **ALICE shows effective sim-to-real transfer, outperforming classical and other zero-shot counterparts**. Here, we showcase ALICE's performance on 4 real-world settings along with 3 synthetic benchmarks. Note that ALICE is only trained on synthetic tables. Regardless, ALICE consistently provides the best AUROC, orientation, and F1 scores across multiple real-world datasets.

a **directed edge-scoring head** uses separate source and destination projections with feature-specific biases to score each ordered feature pair. A **threshold calibrator** is then used to obtain the decoded adjacency matrix. We provide further architectural details in Appendix D.

Primary Training Objective. For every dataset D from the $(D, A) \sim P_{\text{train}}$ pair and for every off-diagonal pair (edge) $\mathbf{E} = \{(i, j) : i \neq j\}$ with $p_{ij} = \text{sigmoid}(L_{ij})$, where L_{ij} is the output from the directed edge-scoring head for features i and j, we use the binary cross-entropy (BCE) loss:

$$\mathcal{L}_{\text{edge}} = -\frac{1}{|\mathbf{E}|} \sum_{(i,j)\in\mathbf{E}} \left[w_{\text{sparse}} A_{ij} \log p_{ij} + (1 - A_{ij}) \log(1 - p_{ij}) \right], \tag{3}$$

where w_{sparse} is the ratio of non-edges to edges, computed from A. We note that w_{sparse} is essential as the graph density can vary widely across the synthetic training distribution.

Additionally, we train the estimator to be invariant to graph-preserving perturbations. **Our key insight** is that training the estimator to be invariant to such perturbations aids in learning graph or causal structure-aware embeddings (Figure 10) and prevents the DAG estimator from developing arbitrary correlations between a specific causal structure and certain mechanisms or noise distributions. Now, we list a few invariance properties and our approach to enforce them in our DAG predictions.

We set up a *width invariance loss* by adding a block of irrelevant columns, and penalize a change in scores. Here we enforce invariance in graph with respect to the width of the table, in particular with respect to the addition of irrelevant columns. The added irrelevant columns should not affect the original graph probabilities. Similarly, when a near copy of a target's parent is added, the edges from the original and duplicate parents to the target are enforced to have equivalent scores, since duplicate columns must be treated identically. Additionally, we also enforce *mechanism invariance*, where from each graph, we generate two observational datasets, each with different edge functions. The predictions corresponding to the datasets of the same graph are enforced to have similar edge score matrices (L), while datasets corresponding to different graphs are encouraged to have dissimilar edge score matrices. Other losses are described in Appendix D.

Similarly, permuting the features in a dataset should leave the predicted causal structure unchanged. However, we note that even without a dedicated loss term, ALICE's *null* mode can achieve permutation invariance as a consequence of the architecture choices, such as shared column processing (θ_{enc}) and variable-wise attention. The same holds true for reordering the i.i.d. observations as a consequence of set-based aggregation (θ_{agg}).

3.3 Invariance-aware Causal Estimation - A Risk Minimization Motivation

We provide a conceptual justification for why invariance-aware training helps. Let θ_{DAG} be the model with the invariance-aware training. We indicate the test distribution that we use to evaluate θ_{DAG} as P_{test}. Similar to P_{train}, we have $(D, A) \sim P_{\text{test}}$, and the task is to estimate A from D.

INVARIANCE	METRIC	ALICE (OURS)	FOUNDCAUSE	CDFM	TABCAUSAL	CAUSCALE	AVICI	PC	DIRECTLINGAM
REALIZATION	AUROC	**0.9397**	0.9058	0.8314	0.8310	0.7018	0.7605	0.6968	0.6622
	CONSISTENCY	0.8204	**0.8210**	0.7348	0.6770	0.6264	0.7231	0.6103	0.5416
MECHANISM	AUROC	**0.9232**	0.8370	0.7974	0.7728	0.7003	0.7491	0.6059	0.6217
	CONSISTENCY	**0.5904**	0.4257	0.3167	0.2473	0.2449	0.2139	0.1973	0.1006
WIDTH	AUROC	**0.9831**	0.9496	0.9302	0.9179	0.8098	0.8637	0.6784	0.6501
	CONSISTENCY	0.8874	0.8627	0.8283	0.6366	0.7637	0.8319	0.6255	**0.9600**
IDENTIFIABILITY	DEVIATION	**0.1129**	0.1631	0.3719	0.3600	0.2955	0.3482	0.4654	0.4616

Table 2: **Enforcing invariance improves ALICE's ability to generate consistent predictions when different observational datasets correspond to the same DAG.** For realization, mechanism, and width invariances, observe that the predictions are correct (high AUROC) and consistent (high Consistency). Consistency is measured using the Spearman rank correlation between the predictions of multiple views. Secondly, we show that ALICE has the least deviation from the expected ideal performance for classical identifiability tests under non-linearity and non-Gaussianity assumptions. Appendix contains the experimental setup (Table 7) and the extended table (Table 8).

Graph-preserving views. For the observational dataset D, let $\mathcal{V}_D$ denote a family of graph-preserving view-generation rules, and let $\mu_{\mathcal{V}_D}$ denote a probability distribution over this family. We sample $v \sim \mu_{\mathcal{V}_D}$ and generate the alternative observational dataset D^v such that the causal structure is unchanged. Each view $v \in \mathcal{V}_D$ has a known invertible transformation ρ_v that transforms A such that $A^v = \rho_v(A)$ is the adjacency matrix corresponding to D^v. For example, $D^v = f_v(D)$ for deterministic transformations such as permutations ($\rho_v(A) = P_\pi A P_\pi^\top$ with P_π representing the standard permutation matrix for the permutation π).

Assumption 3.1 (Distribution-preserving view generation)**.** For an observational dataset D and its causal graph A sampled from P_{test}, let $v \sim \mu_{\mathcal{V}_D}$ be a graph-preserving view such that it transforms D to D^v with the corresponding causal graph being $A^v = \rho_v(A)$. The resulting dataset-graph pair follows the same distribution as the original pair:

$$(D, A) \overset{d}{=} (D^v, \rho_v(A)) \,. \tag{4}$$

Here, equality in distribution is taken over the view selection and generation.

Assumption 3.2 (Invariant convex loss)**.** For any view $v \sim \mu_{\mathcal{V}_D}$ and its associated invertible transformation ρ_v, the causal-structure loss is invariant when both prediction $\hat{A}$ and target A are transformed with ρ_v:

$$\ell\Big(\rho_v(\hat{A}), \rho_v(A)\Big) = \ell(\hat{A}, A). \tag{5}$$

Additionally, ℓ is nonnegative and for a given target graph A, $\ell(\cdot, A)$ is convex in its prediction.

Now, we list the views and their required conditions to qualify for Assumption 3.1. For **permutation views**, we require a dataset-graph pair in $(D, A) \sim P_{\text{test}}$ to be obtained from any possible permutation of a dataset-graph pair from the original distribution $P_{\text{test}}^{\text{raw}}$. This condition allows for distribution preservation when the view is a random permutation. Importantly, note that this requirement poses no restrictions on the graph structures, mechanisms, or noise distributions. We provide a detailed discussion in Appendix B.1 along with the conditions for **mechanism views**. Assumption 3.2 holds for the standard and sparsity-weighted BCE loss, since it is convex in the predicted edge probabilities. Moreover, applying ρ_v jointly to the prediction and target only changes the indices of the corresponding edge entries, and therefore the BCE remains invariant.

Now, we define an aggregate predictor as $\hat{A}_{\text{agg}} = \mathbb{E}_{v \sim \mu_{\mathcal{V}_D}}\left[\rho_v^{-1} \theta_{\text{DAG}}(D^v)\right]$. Consequently, the expected risks of the original and the aggregate predictor are given by:

$$R(\theta_{\text{DAG}}) = \mathbb{E}_{(D,A)\sim P_{\text{test}}}\left[\ell(\theta_{\text{DAG}}(D), A)\right] \qquad R_{\text{agg}}(\theta_{\text{DAG}}) = \mathbb{E}_{(D,A)\sim P_{\text{test}}}\left[\ell\Big(\hat{A}_{\text{agg}}, A\Big)\right]. \tag{6}$$

Theorem 3.3 (Risk monotonicity under multi-view aggregation)**.** *For a given causal estimator θ_{DAG} and a corresponding multi-view aggregate predictor $\hat{A}_{\mathrm{agg}}$ with their estimation risks $R(\theta_{\mathrm{DAG}})$ and $R_{\mathrm{agg}}(\theta_{\mathrm{DAG}})$ as defined in Equation 6, if Assumptions 3.1 and 3.2 hold true, then we have*

$$R_{\mathrm{agg}}(\theta_{\mathrm{DAG}}) \leq R(\theta_{\mathrm{DAG}}). \tag{7}$$

We provide the proof of Theorem 3.3 in Appendix B, which follows from Jensen's inequality. In essence, the theorem establishes that multi-view aggregation cannot increase the expected causal-structure risk under Assumptions 3.1 and 3.2.

In general, ALICE and the theoretical justification for aggregation *do not* make an observationally unidentifiable causal direction identifiable. In particular, when multiple DAGs belong to the same Markov equivalence class and cannot be distinguished from the available observational distribution, multi-view aggregation cannot by itself resolve this ambiguity. The theorem only states that predictions should not vary under transformations that preserve causal information.

3.4 From Aggregation to Test-Time Improvements

In this section, we propose test-time approaches that help improve the overall graph estimation. **Our key practical insight** from Theorem 3.3 is that averaging ALICE's predictions across multiple graph-preserving views can reduce view-specific variance. Inspired by this, we propose two test-time improvement algorithms.

Target Averaging is a strategy where we assign a random feature as the target. For a given dataset D, from Equation 17, note that as the target changes, the token representations for all features change. Intuitively, for target t, we can equivalently associate the change in the feature representations to a new DAG estimator θ^t_{DAG}. Therefore, for target averaging, we define the following aggregate predictor: $\hat{A}_{\mathrm{agg}} = \mathbb{E}_t[\theta^t_{\mathrm{DAG}}(D)]$. To this end, for a given dataset-graph pair (D, A), the per-sample risk for $\hat{A}_{\mathrm{agg}}$ is given by $\ell(\mathbb{E}_t[\theta^t_{\mathrm{DAG}}(D)], A)$. Using Assumption 3.2, since ℓ is convex in its predictions, we directly get $\ell(\mathbb{E}_t[\theta^t_{\mathrm{DAG}}(D)], A) \leq \mathbb{E}_t[\ell(\theta^t_{\mathrm{DAG}}(D), A)]$. In other words, for a dataset D, target averaging has no larger loss than the expected loss of a randomly selected target view.

Subgraph Voting is a strategy where we repeatedly sample a subset $S \subseteq \{1, \ldots, N_{\mathrm{feat}}\}$ of features from the dataset D and apply the DAG estimator to the corresponding sub-dataset D_S. For a given ordered feature pair (i, j), let $\mathcal{S}_{ij}$ denote the set of sampled subgraphs that contain both i and j. Note that the edge existence is preserved in all subgraphs, *i.e.,* $(A_S)_{ij} = A_{ij}$ for every $S \in \mathcal{S}_{ij}$. Therefore, we can define a per-edge aggregated predictor, and following the line of proof as similar to Target Averaging, we can show that for each edge, averaging predictions over multiple subgraphs gives a loss that is no larger than the average loss from a randomly sampled subgraph containing that edge, *i.e.,* $\ell_{ij}(\mathbb{E}_{S\in\mathcal{S}_{ij}}[\theta^S_{\mathrm{DAG}}(D_S)_{ij}], A_{ij}) \leq \mathbb{E}_{S\in\mathcal{S}_{ij}}[\ell_{ij}(\theta^S_{\mathrm{DAG}}(D_S)_{ij}, A_{ij})]$.

Note that our discussion does not support the following two arguments: 1. Subgraph-aggregated single edge prediction loss is not greater than its full graph counterpart; 2. Target-averaged prediction loss is not greater than null mode's prediction loss. Instead, we provide strong empirical evidence and intuitive explanations for these behaviors in Section 4.

Overall, for test-time improvements, ALICE combines accurate single-pass graph estimation (Table 1) with improved consistency (Table 2) across graph-preserving views. From standard Monte Carlo estimation results (Haghighat and Wagner, 2003), the variance of a finite-sample average decreases with the number of samples, and estimators with lower view-induced variation require fewer samples to accurately approximate their population average ($\hat{A}_{\mathrm{agg}}$). Thus, **the improved prediction consistency enforced by our invariance-aware training can make test-time aggregation more compute-efficient**. We show this in Figure 3, where CDFM and TabCausal struggle to obtain aggregation gains with inconsistent predictions, while ALICE's consistent predictions provide up to 14% gains in F1 score.

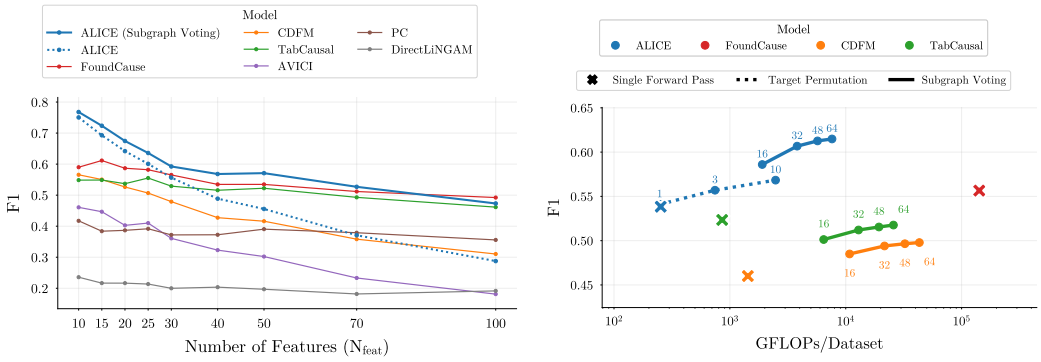


Figure 3: **Invariance-aware training enables effective and compute-efficient test-time aggregation. (Left)** With increasing number of features, the F1 score degrades for ALICE with single forward pass. However, Subgraph Voting mitigates this by decomposing wide graphs into smaller inference problems. **(Right)** Subgraph Voting significantly improves performance with limited additional compute. We attribute this to consistent predictions enforced by invariance-aware training.

4 Experimental Results

We first describe the evaluation procedure. ❶ **Datasets:** We evaluate ALICE on synthetic and real-world CD benchmarks. For synthetic evaluation, we generate held-out SCMs (Section 3.1), varying the sample size and number of features. We also construct paired synthetic benchmarks to evaluate invariance properties (see Table 2). For real-world evaluation, we use four benchmarks: **Causal Chambers** (Gamella et al., 2025), **Tübingen Cause-Effect Pairs** (Mooij et al., 2016), **Sachs** (Sachs et al., 2005), and **CausalRivers** (Stein et al., 2025). ❷ **Metrics:** Since ALICE predicts a probability score for every ordered pair (i, j), the Area Under the Receiver Operating Characteristic curve metric (AUROC) measures how well these scores rank true directed edges above non-edges over the complete adjacency matrix, without requiring a fixed decision threshold. As per Tübingen's, protocol, we report weighted orientation accuracy using the provided weights for cause-effect pairs. For graph recovery estimation, we report the F1 scores. For the invariance experiments, we measure both accuracy (edge AUROC) and *consistency* (Spearman rank correlation between the directed edge-score rankings obtained from the two views). ❸ **Baselines:** We compare ALICE against both amortized CD models and established dataset-specific causal-discovery algorithms. For amortized approaches, we consider **CDFM**, **TabCausal**, **FoundCause**, **CauScale** (Peng et al., 2026) and **AVICI** (Lorch et al., 2022). We additionally compare against classical approaches, such as **PC** and **DirectLiNGAM**.

Despite being trained only on synthetic SCMs, **ALICE generalizes strongly to real-world CD tasks** with diverse data generating processes (Table 1). Our experiments reveal the following key insights.

Invariance loss terms enable ALICE to extract the underlying DAG structure. Table 9 in the Appendix shows the performance of ALICE in the presence and absence of invariance losses. Observe that for both single-view and aggregated estimation, ALICE with invariance losses provides highly accurate and consistent predictions when compared to the case without invariance losses. These results indicate that the model preserves its edge-ranking structure across causally equivalent views rather than relying on realization- or mechanism-specific artifacts (Figure 10).

The target-aware mode infers better causal relationships than the null mode. We hypothesize that adding the target embedding to all feature embeddings makes it easier for ALICE to recover causal relationships involving the selected target and features in its vicinity, while leaving estimation of distant inter-feature relationships largely unchanged. Figure 6a supports this hypothesis on graphs constructed from three disjoint local subgraphs. Selecting a target from one subgraph primarily improves recovery within that subgraph, whereas performance in the other two remains close to the null mode. Different target views therefore provide complementary local gains, and averaging their predictions produces a more accurate estimate of the complete graph. Accordingly, target-aware single-pass inference outperforms null inference once the graph moves beyond the smallest-width settings in Figure 6b. However, target averaging remains limited by graph width because every target-conditioned pass must still estimate the complete graph, causing its localized advantage to become diluted as the number of features grows.

Subgraph voting overcomes this limitation by *decomposing a wide graph into smaller inference problems*, where target-aware inference remains effective, and aggregating these locally accurate predictions into a stronger and more stable global graph estimate. This is empirically shown through F1 gains (15% gains) in Figure 3, and Figure 9 clearly shows advantage of decomposing a wide graph into smaller inference problems. Interestingly, **Subgraph Voting achieves these gains while remaining compute-efficient.** We attribute this compute-efficiency to the invariance-aware training, which improves ALICE's prediction consistency across different views of the same DAG (Table 2). ALICE acts as an accurate low-variance predictor, thereby requiring fewer samples to accurately approximate their aggregate estimator ($\hat{A}_{\text{agg}}$). Overall, Subgraph Voting is not specific to ALICE. In Figure 7, we extend it to CDFM and TabCausal. Interestingly, for all 3 models, the improvements saturate beyond 30 subgraph features, further validating the decomposition claim.

Feature semantics provide complementary evidence for real-world causal discovery that can be used to improve ALICE's predictions. We incorporate this information through a simple post-processing rule applied to the predicted edge probabilities $p_{ij}^{\theta_{\text{DAG}}} = \text{sigmoid}(L_{ij})$ defined earlier. We pass in column names and prompt for an edge confidence score for every $i \to j$, which ensures that the information received by the LLM is completely disjoint and orthogonal to the information received by θ_{DAG}. This allows us to assume that the LLM prediction of an edge score, p_{ij}^{LLM} is an independent estimate of probability to that of the numerical model. We predict a joint estimate of edge probability $p_{ij} = 1 - (1 - p_{ij}^{\theta_{\text{DAG}}})(1 - p_{ij}^{\text{LLM}})$, that is, an OR operation of the two model predictions. As shown in Figure 8, this fixed fusion improves Tübingen weighted orientation accuracy 18.6% from 0.63 to 0.75 for ALICE, 53.0% for FoundCause, along with improvements on the other models as well. On the Intersection Test (see Figure 4), described in Appendix E, edge AUROC improves 29.6% for TabCausal. We do not attempt this exercise on Causal Chambers, where semantics are less informative than the highly accurate numerical predictions. In all datasets tested, this method improves or retains the same level of performance.

ALICE effectively recognizes when causal direction becomes identifiable, particularly as relationships depart from linearity. Across the nonlinearity sweep in the identifiability analysis (see Appendix F and Figure 5), ALICE closely follows the ideal transition from chance-level accuracy at the linear-Gaussian boundary to near-perfect direction recovery in identifiable regimes, achieving the lowest deviation from ideal behavior among the foundation models evaluated. It also remains competitive across changes in non-Gaussianity. Although the invariance losses in pretraining do not directly supervise identifiability, they appear to make ALICE more sensitive to the conditions under which causal direction is statistically identifiable (Table 9).

Limitations. ALICE has three important limitations. First, observational data alone cannot, in general, uniquely orient edges within a Markov equivalence class. This is a well-established problem in the causal discovery literature, where the directions cannot be estimated without additional assumptions. Therefore, ALICE remains uncertain or incorrect on such directions. Second, although the model predicts directed edge scores, it does not explicitly guarantee an acyclic output, and we require post-processing to remove residual cycles. Finally, ALICE estimates the causal graph structure but does not recover the underlying functional relationships associated with the edges.

5 Conclusion and Future Work

We introduced **ALICE**, an amortized causal discovery model that estimates full directed causal graphs from observational data in a single forward pass and generalizes from synthetic pretraining to diverse real-world settings. Additionally, we enforce invariances through targeted augmentations and consistency objectives, improving both graph recovery and prediction stability across equivalent views of the same underlying causal structure. Building on this, we further introduced test-time multi-view aggregation strategies and a simple semantic post-processing approach that incorporates feature-level textual information to improve causal estimation on real-world datasets.

An important direction for future work is to extend ALICE beyond graph structure estimation to also recover the underlying structural functions associated with each causal edge. In addition, ALICE could be integrated with tabular and multivariate time series foundation models, where the estimated causal structure can provide an explicit structural prior for downstream regression and forecasting.

6 AI USE STATEMENT

In this work, we used generative AI tools to assist in implementing our method and experiments. We also used generative AI tools to assist in the verification of proofs. We did not use it to build our theoretical/conceptual framework, formulate mathematical claims, provide critical ingredients for proving mathematical claims, refine our hypotheses, generate synthetic data, or support qualitative/thematic data analysis and interpretation of results. We also did not use AI tools to clean, reformat, or otherwise modify any dataset. However, we did use AI tools to refine the written content, edit existing codebases, summarize existing literature, and identify literature relevant to our method. We have reviewed the code produced and proofs verified with AI assistance and take responsibility for the final content of this work, including the text, claims, and artifacts produced with the aid of generative AI.

7 ETHICS STATEMENT

This work studies zero-shot causal discovery from observational data and raises no concerns we are aware of under the Code of Ethics. No human subjects took part in this research. All pretraining data are synthetically generated from structural causal models, and the benchmarks we evaluate on (Causal Chambers, Tübingen cause–effect pairs, Sachs protein-signaling data, CausalRivers, CausalArena) are existing public releases used under their respective licenses, so no new personal or sensitive data was collected. We do not release any new dataset containing personal information. All datasets used contain only numerical or categorical feature values, sensor readings, or protein-expression measurements. The primary risk we are aware of is the potential misuse of predicted causal graphs as confirmed causal claims in high-stakes domains (e.g., biology, finance, policy) without further domain-expert validation. As discussed in Section 4, ALICE does not resolve the fundamental non-identifiability of causal direction within a Markov equivalence class, and its predictions should be treated as hypotheses for further testing rather than ground truth. The GPT-OSS-120B model used for the feature-semantics post-processing experiments was accessed under its public release terms. The authors declare no conflicts of interest and no sponsorship that bears on the results.

8 REPRODUCIBILITY STATEMENT

To help reproduce our results, we provide exhaustive details about our training corpus generation method in Appendix C, including details about graph sampling, root mechanisms, noise distributions, sample rejection criteria and episode types we use.

We provide a detailed description of our architecture in Appendix D. This includes our choice of hyperparameters, an exact description of our architecture components, and the reasoning behind our design choices.

Finally, in Section 4 and Appendix I we describe in detail our model performance across a suite of metrics. Due to the stochastic nature of some of these models, the reviewer may only be able to make an reproduction of our numbers up to two significant figures.

In addition, we intend to open source the checkpoint, model code, and evaluation datasets upon acceptance.

REFERENCES

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting, 2025. URL https://arxiv.org/abs/2510.15821.

Kevin Bello, Bryon Aragam, and Pradeep Ravikumar. Dagma: Learning dags via m-matrices and a log-determinant acyclicity characterization. *Advances in Neural Information Processing Systems*, 35:8226–8239, 2022.

Patrick Blöbaum, Krishnakumar Balasubramanian, and Shiva Prasad Kasiviswanathan. Foundcause: Causal discovery with latent confounders from observational data, 2026. URL https://arxiv.org/abs/2606.17516.

Stephan Bongers, Patrick Forré, Jonas Peters, and Joris M Mooij. Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885–2915, 2021.

Philippe Brouillard, Sébastien Lachapelle, Alexandre Lacoste, Simon Lacoste-Julien, and Alexandre Drouin. Differentiable causal discovery from interventional data. *Advances in Neural Information Processing Systems*, 33:21865–21877, 2020.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554, 2002.

Ben Cohen, Emaad Khwaja, Youssef Doubli, Salahidine Lemaachi, Chris Lettieri, Charles Masson, Hugo Miccinilli, Elise Ramé, Qiqi Ren, Afshin Rostamizadeh, et al. This time is different: An observability perspective on time series foundation models. *Advances in neural information processing systems*, 38:50907–50951, 2026.

Juan L Gamella, Jonas Peters, and Peter Bühlmann. Causal chambers as a real-world physical testbed for ai methodology. *Nature Machine Intelligence*, 7(1):107–118, 2025.

Clive WJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, pages 424–438, 1969.

Alireza Haghighat and John C Wagner. Monte carlo variance reduction with deterministic importance functions. *Progress in Nuclear Energy*, 42(1):25–53, 2003.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13(1): 2409–2464, 2012.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

Nan Rosemary Ke, Silvia Chiappa, Jane X. Wang, Jörg Bornschein, Anirudh Goyal, Mélanie Rey, Theophane Weber, Matthew M. Botvinick, Michael Curtis Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. In *ICLR*, 2023. URL https://openreview.net/forum?id=hp_RwhKDJ5.

Zi-Rong Li, Si-Yang Liu, Tian-Zuo Wang, and Han-Jia Ye. Causalarena: Benchmarking causal discovery in the foundation model era, 2026a. URL https://arxiv.org/abs/2609.11897.

Zi-Rong Li, Si-Yang Liu, Tian-Zuo Wang, and Han-Jia Ye. Tabcausal: Pretraining across causal environments for tabular causal discovery, 2026b. URL https://arxiv.org/abs/2605.31156.

Lars Lorch, Scott Sussex, Jonas Rothfuss, Andreas Krause, and Bernhard Schölkopf. Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35: 13104–13118, 2022.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Conference on causal learning and reasoning*, pages 509–525. PMLR, 2022.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Scalable causal discovery with score matching. In *Conference on Causal Learning and Reasoning*, pages 752–771. PMLR, 2023.

Joris M Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *Journal of Machine Learning Research*, 17(32):1–102, 2016.

Bo Peng, Sirui Chen, Jiaguo Tian, Yu Qiao, and Chaochao Lu. Causcale: Neural causal discovery at scale. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=UCoD9hpiJs.

Jie Qiao, Ruichu Cai, Zijian Li, Weilin Chen, Pengfei Hua, Boyan Xu, Zhengming Chen, Zhifeng Hao, and Peng Cui. Cdfm: Towards a general-purpose causal discovery foundation model, 2026. URL https://arxiv.org/abs/2607.11508.

Jingang QU, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=0VvD1PmNzM.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D Mahecha, Jordi Muñoz-Marí, et al. Inferring causation from time series in earth system sciences. *Nature communications*, 10(1):2553, 2019.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005.

Agathe Sadeghi, Achintya Gopal, and Mohammad Fesanghary. Causal discovery in financial markets: A framework for nonstationary time-series data. *arXiv preprint arXiv:2312.17375*, 2023.

Muralikrishnna G Sethuraman, Romain Lopez, Rahul Mohan, Faramarz Fekri, Tommaso Biancalani, and Jan-Christian Hütter. Nodags-flow: Nonlinear cyclic causal structure learning. In *International Conference on Artificial Intelligence and Statistics*, pages 6371–6387. PMLR, 2023.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, Antti Kerminen, and Michael Jordan. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. The MIT press, 2001.

Gideon Stein, Maha Shadaydeh, Jan Blunk, Niklas Penzel, and Joachim Denzler. Causalrivers-scaling up benchmarking of causal discovery for real-world time-series. In *International Conference on Learning Representations*, volume 2025, pages 64516–64531, 2025.

Ryan Thompson, He Zhao, Daniel M. Steinberg, and Edwin V. Bonilla. Arrow: A Foundation Model for Causal Discovery, May 2026. URL http://arxiv.org/abs/2605.07204. arXiv:2605.07204 [cs.LG].

Yuanrui Wang, Xingxuan Zhang, Han Yu, Mingchao Hao, Gang Ren, Hao Yuan, Li Mao, Yunjia Zhang, Chun Yuan, and Peng Cui. Limix-2m: Mitigating low-rank collapse and attention bottlenecks in tabular foundation models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=YjYPYg112A.

Menghua Wu, Yujia Bao, Regina Barzilay, and Tommi Jaakkola. Sample, estimate, aggregate: A recipe for causal discovery foundation models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=h434zx5SX0.

Jiaqi Zhang, Kristjan Greenewald, Chandler Squires, Akash Srivastava, Karthikeyan Shanmugam, and Caroline Uhler. Identifiability guarantees for causal disentanglement from soft interventions. *Advances in Neural Information Processing Systems*, 36:50254–50292, 2023.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric Xing. Learning sparse nonparametric dags. In *International conference on artificial intelligence and statistics*, pages 3414–3425. Pmlr, 2020.

Appendix Overview

A Extended Related Works **15**

B Proof of Theorem 3.3 **16**

B.1 Assumption Validation 16

C Synthetic Training Corpus **17**

D Detailed Description of Architecture and Training **19**

E Simulating Traffic Incidents with Counterfactual Causal Ground Truth **22**

F Identifiability Analysis **23**

G Evaluation Metrics **24**

H Evaluation Datasets **25**

H.1 Synthetic Datasets 25

H.2 Invariance Datasets 26

H.3 Real-World Datasets 27

I Extended Results **27**

A Extended Related Works

Statistical and Classical Causal Discovery. Classical CD recovers the structure of the graph from conditional independence tests or from score optimization under assumptions about the generating mechanism. Constraint-based methods, such as the Peter-Clark (PC) algorithm (Spirtes et al., 2001), first recover an undirected skeleton through independence testing, then apply orientation rules, returning a Markov equivalence class rather than a single DAG. Score-based methods such as GES (Chickering, 2002) search greedily over equivalence classes, adding and deleting edges to improve a penalized likelihood. Functional approaches instead restrict the structural equations: Shimizu et al. (2006) proposed LiNGAM, which assumes linear mechanisms with non-Gaussian noise, and additive noise models (ANM) (Hoyer et al., 2008) assume nonlinear mechanisms with additive noise, in both cases breaking the symmetry between cause and effect so that direction becomes identifiable. For time series, Granger causality (Granger, 1969) tests whether the lagged values of one variable improve the prediction accuracy of another. These statistical methods, while drawn from first principles, often make linearity, Gaussian noise or time invariant assumptions about the data that make them impractical to apply on real world datasets with complex time- or row-varying relationships between variables or non-linearity.

Later works incorporate neural techniques to avoid data assumptions, re-framing structure learning as a differentiable problem that avoids explicit combinatorial search: NOTEARS (Zheng et al., 2018; 2020) introduced a smooth algebraic characterization of acyclicity based on the trace of a matrix exponential, turning DAG learning into gradient-based optimization. Subsequent work tests alternative acyclicity characterizations (Bello et al., 2022; Montagna et al., 2023), incorporates interventional data (Brouillard et al., 2020), or abandons acyclicity altogether in favor of cyclic models (Sethuraman et al., 2023). What all of these share is that each dataset requires solving a fresh optimization problem from scratch.

Our proposed algorithm instead performs pre-training over large-scale synthetic data, and demonstrates zero-shot generalization that beats these single dataset methods without fine-tuning.

Foundation Models for Regression and Classification. Foundation models for structured data recast prediction as zero-shot in-context inference backed by large-scale pretraining. On tabular data, models such as TabPFN, LimiX, and TabICL (Hollmann et al., 2025; Wang et al., 2026; QU et al., 2025) are pretrained on large synthetic or heterogeneous corpora and predict a target column in a single forward pass, with no task-specific fine-tuning. The same recipe has been used to forecast multivariate time series (Ansari et al., 2025; Cohen et al., 2026), where a single pretrained model forecasts unseen series zero-shot, contextualizing each variate against the others rather than fitting a separate model per series. In both settings the pretraining distribution assumes the role that a hand-specified model class plays in classical approaches, and in the tabular case that distribution is itself generated from structural causal models, so the pretrained predictor is already exposed to data carrying causal structure.

The prediction target, however, remains a labeled column or a future window rather than the relationships between the variables themselves, so strong predictive performance does not carry over to causal discovery directly.

General Neural Models for Causal Discovery. Causal discovery foundation models apply the same pretraining recipe to the structural problem, mapping a tabular dataset to edge probabilities or an adjacency matrix in one pass. Early amortized methods established the template, training neural predictors on synthetic graphs and mechanisms to map observational and interventional samples directly to structure (Lorch et al., 2022; Ke et al., 2023), or to time-series data by separating shared dynamics from a per-sample graph encoder (Löwe et al., 2022). Wu et al. (2025) instead train a supervised aggregator over summary statistics and the outputs of classical algorithms run on sampled variable subsets, rather than predicting edges from raw values.

TabCausal (Li et al., 2026b) argues that the binding constraint on such models is how the pre-training tasks are constructed, and pre-trains across a broad space of graph priors, mechanism families, noise models, dimensions, sample sizes, and observational versus mixed-interventional regimes, consuming per-cell intervention indicators alongside the observed values. CDFM (Qiao et al., 2026) approaches the problem from the identifiability side: it treats the unknown mechanism as a latent variable and

decomposes the marginal likelihood into a variational objective over mechanism inference, data reconstruction, and graph decoding, instantiated as alternating row–column attention, a masked-imputation head, and a mixture-of-experts bilinear edge decoder. FoundCause (Blöbaum et al., 2026) operates purely observationally and adds a dedicated confounder module built on learnable latent representations, predicting a symmetric confounding matrix alongside the directed graph. Concurrent efforts target scale and architecture rather than pretraining coverage (Thompson et al., 2026; Peng et al., 2026).

Existing causal discovery methods, however, do not fully exploit the invariance properties of the underlying causal graph of an observed dataset. We demonstrate that by including this information we are able to improve significantly upon their results across a variety of domains.

B Proof of Theorem 3.3

Under Assumptions 3.1 and 3.2, we have to prove $R_{\text{agg}}(\theta_{\text{DAG}}) \leq R(\theta_{\text{DAG}})$. However, from Equation 6, we have

$$R_{\text{agg}}(\theta_{\text{DAG}}) = \mathbb{E}_{(D,A)\sim P_{\text{test}}}\left[\ell\Big(\mathbb{E}_{v\sim\mu_{\mathcal{V}_D}}\left[\rho_v^{-1}\theta_{\text{DAG}}(D^v)\right], A\Big)\right]. \tag{8}$$

From Assumption 3.2, ℓ is convex in the prediction argument. Therefore, using the Jensen's inequality, we have

$$R_{\text{agg}}(\theta_{\text{DAG}}) \leq \mathbb{E}_{(D,A)\sim P_{\text{test}}}\mathbb{E}_{v\sim\mu_{\mathcal{V}_D}}\left[\ell(\rho_v^{-1}\theta_{\text{DAG}}(D^v), A)\right]. \tag{9}$$

Using the invariance property of the loss from Assumption 3.2, we have $\ell(\rho_v^{-1}\theta_{\text{DAG}}(D^v), A) = \ell(\theta_{\text{DAG}}(D^v), \rho_v A)$. Therefore, we get

$$R_{\text{agg}}(\theta_{\text{DAG}}) \leq \mathbb{E}_{(D,A)\sim P_{\text{test}}}\mathbb{E}_{v\sim\mu_{\mathcal{V}_D}}\left[\ell(\theta_{\text{DAG}}(D^v), \rho_v(A))\right]. \tag{10}$$

Using the equality in distribution assumption from Assumption 3.1, $(D^v, \rho_v(A))$ and (D, A) are equal in distribution. Therefore, the expected loss over the transformed distribution and the expected loss over the original distribution are equal, ***i.e.,***

$$\mathbb{E}_{(D,A)\sim P_{\text{test}}}\mathbb{E}_{v\sim\mu_{\mathcal{V}_D}}\left[\ell(\theta_{\text{DAG}}(D^v), \rho_v(A))\right] = \mathbb{E}_{(D,A)\sim P_{\text{test}}}\left[\ell(\theta_{\text{DAG}}(D), A)\right]. \tag{11}$$

Consequently, we use this in Equation 10 to get

$$\begin{aligned} R_{\text{agg}}(\theta_{\text{DAG}}) &\leq \mathbb{E}_{(D,A)\sim P_{\text{test}}}\left[\ell(\theta_{\text{DAG}}(D), A)\right], &(12)\\ &= R(\theta_{\text{DAG}}). &(13)\end{aligned}$$

Therefore, we have $R_{\text{agg}}(\theta_{\text{DAG}}) \leq R(\theta_{\text{DAG}})$.

B.1 Assumption Validation

Here, we describe the condition required for mechanism and permutation views to qualify for Assumption 3.1 in detail.

Permutation views. Consider an arbitrary distribution of original test pairs $(D_0, A_0) \sim P_{\text{test}}^{\text{raw}}$. For a pair with N_{feat} features, we sample $q \sim \text{Unif}(\mathcal{S}_{N_{\text{feat}}})$, where $\mathcal{S}_{N_{\text{feat}}}$ is the set of all feature permutations, and construct $(D, A) = (D_0 P_q^\top, P_q A_0 P_q^\top) \sim P_{\text{test}}$. Now consider any additional permutation $r \in \mathcal{S}_{N_{\text{feat}}}$. Applying this view gives $(DP_r^\top, P_r A P_r^\top)$. Since q is uniformly distributed, the composition $r \circ q$ is also uniformly distributed over $\mathcal{S}_{N_{\text{feat}}}$. Therefore, $(DP_r^\top, P_r A P_r^\top) \overset{d}{=} (D, A)$, satisfying Assumption 3.1.

Mechanism views. Let $\mathcal{H}$ denote a family of invertible feature-wise transformations and let P_H denote a distribution over this family. To obtain $(D, A) \sim P_{\text{test}}$, we start from $(D_0, A_0) \sim P_{\text{test}}^{\text{raw}}$, sample N_{feat} functions from H, *i.e.,* $H = (h_1, \ldots, h_{N_{\text{feat}}}) \sim P_H$, and construct $D_{:,j} = h_j((D_0)_{:,j})$ and $A = A_0$. Thus, the feature values and their functional relationships can change while the underlying causal graph remains unchanged. For an additional mechanism view $H' \sim P_H$, Assumption 3.1 requires $H' \circ H \overset{d}{=} H$, which ensures that $(H'(D), A) \overset{d}{=} (D, A)$. Here, $\circ$ represents the composition of the feature-level functions. For example, for the first feature, we have the condition $h_1'(h_1(D_{0_{:,1}})) \overset{d}{=} D_{:,1}$. Therefore, the condition for mechanism views to satisfy Assumption 3.1 is the following: P_H must be chosen such that applying another admissible transformation preserves the test distribution.

C Synthetic Training Corpus

The pretraining distribution P_{SCM} is sampled hierarchically. An SCM $\mathcal{M} = (\mathbf{g}, P_\epsilon)$ is drawn by first sampling its graph and then its structural functions and noise distribution, after which a dataset is drawn from it:

$$A \sim P_{\text{graph}}, \qquad \mathcal{M} \sim P_{\text{SCM}}(\cdot \mid A), \qquad D \sim P_{\mathcal{M}}, \tag{14}$$

where $A \in \mathcal{A}_{N_{\text{feat}}}$ and $D \in \mathbb{R}^{N_{\text{obs}} \times N_{\text{feat}}}$. Recall from Section 2 that each feature is a structural function of its parents, $X_i = g_i(X_{\mathbf{pa}_A(i)}, \epsilon_i)$; g_i combines a deterministic mechanism $f_i : \mathbb{R}^{|\mathbf{pa}_A(i)|} \to \mathbb{R}$ of the parents with the noise ϵ_i. Several families build f_i from univariate *edge functions* $f_{ji} : \mathbb{R} \to \mathbb{R}$, one for each parent $j \in \mathbf{pa}_A(i)$. Features are simulated in topological order. Parents are standardized before entering f_i, and every generated feature is standardized; we write $\text{stdz}(\cdot)$ for standardization. Draws with non-finite or near-constant columns are rejected and resampled. Most episodes are row-exchangeable *cross-sectional* samples from Equation (14). The remaining *temporal* and *interventional* episodes are described at the end of the section. Episodes are generated on the fly, so no dataset is seen twice.

Graph Structure. P_{graph} is a mixture over the families in Table 3. Graph density is sampled based on structure. A uniformly random permutation π acts jointly as $(D, A) \mapsto (DP_\pi^\top, P_\pi A P_\pi^\top)$, so column position carries no information about causal order.

Roots. Roots are noise drawn from the families in Table 4. We use different families for time series and tabular synthetic data, and in particular noise is I.I.D between rows in tabular but not in time series data.

Mechanisms. We describle in Table 5 the families from which edge mechanisms are drawn. Across all families, the corpus contains smooth and discontinuous, additive and non-additive, and exact and noisy mechanisms, over continuous, integer, ordinal and categorical variables.

Noise and Observation Transforms. Homogeneous SCMs draw one noise family for the whole dataset, and the other SCMs draw one per node. The noise families are Gaussian, uniform, Laplace, clipped Cauchy, Student-t, Beta, exponential, lognormal, Gumbel and Gaussian mixtures. After generation, a dataset can be stressed by monotone column warping, sparse outliers, mean imputation, or permuting a few entries within a column. These stressors act only on D and leave A unchanged, and discrete columns are exempt from continuous perturbations.

Temporal episodes. In a time series, rows are not exchangeable and causes act with a delay. Temporal episodes use small, weakly connected graphs and one of two families:

Distributed-lag series. Root series follow the time-series processes in Table 4, whose varied dynamics reveal nothing about a series' causal role. A child series i is built from each parent j in four steps.

(i) Response. a random pointwise function, such as a square, $\tanh$, threshold step or first difference, lets the child react to its parent's level, magnitude, threshold crossings or changes.

(ii) Onset delay and distributed-lag kernel: allow an effect to arrive over multiple future steps,

Group	Family	Structure
Random	Erdős–Rényi	independent edges at a sampled density
	Scale-free	preferential attachment, with hubs as causes or as effects
	Watts–Strogatz	rewired ring lattice (small world)
	Stochastic block	dense communities with sparse links between them
	Random geometric	nodes linked within a radius in the unit square
	Layered	edges only between adjacent layers
	Tree, polytree	recursive tree; random orientation creates colliders
	Two-family overlay	two families on a random node split, plus forward edges
Motif	Empty	no edges
	Chain	a single directed path
	Star	one root causing every other node
	Complete	every forward edge
	Fan	shared parents over children with private parents
Sparse deep	Skip chain	a long path plus short-span shortcuts
	Recency-biased tree	tree attaching to recent nodes, so paths are deep
	Bipartite-layered	two or three stages with many roots
	Block-sparse layered	layered blocks with sparse forward cross-block edges
	Dense local modules	dense blocks with sparse links between blocks
	Collider-rich	a sparse base with planted v-structures
Empirical	Regulatory subnetwork	real biological regulatory subgraphs, with hub-and-cascade structure
Planted	Instrumented bipartite	shared and private roots feeding many sinks
	Grouped instrumented	sink banks with weak local edges, weaker than sibling correlations
	Cause-sparse	few causes among distractors, plus near-deterministic descendants

Table 3: **Graph families in the support of P_{graph}** Planted families contain hard configurations: weak true edges, confounded siblings, and target descendants more predictive than its causes.

Family	Type	Description
I.I.D.	Symmetric	uniform, Gaussian, truncated normal, Laplace, Student-t, clipped Cauchy
	Skewed	lognormal, gamma, scaled Beta, exponential, Gumbel
	Multimodal	Gaussian mixtures with random weights, means and scales
	Binary and count	Bernoulli, Poisson
	Categorical	ordinal, multinomial, truncated Zipf
	Jointly dependent	one-factor Gaussian, low-rank Gaussian, uniform ball
Time series	Random walk	non-stationary drift
	Trend	a linear trend plus noise: deterministic drift
	Autoregressive	persistent or quasi-periodic
	Seasonal	a sum of sinusoids plus noise: periodic
	Regime-switching	piecewise-constant levels plus noise: abrupt level shifts
	Spiky	sparse random shocks that decay geometrically: bursty
	Chaotic	logistic map: irregular
	White noise	no memory

Table 4: **Root distributions** I.I.D. families generate the roots of cross-sectional SCMs; jointly dependent roots are drawn as one block.

(iii) Weighting and aggregation. a signed weight lets each parent excite or inhibit the child, and a sum, maximum or gate, allows non-additive interactions between parents.

(iv) Own dynamics and noise. a gain sets how visible the parents are, which, keeps the child from being a filtered copy of its parents.

Vector autoregression. A linear, single-lag relation updates the row vector $\mathbf{X}(t)$ of all series jointly. These series are presented as ordinary tables of time steps, where $j \to i$ shows within a row since $X_i(t)$ depends on $X_j(t-1)$, which is correlated with $X_j(t)$.

Lag tables. Each distributed-lag episode is converted into a table. For a target series j, every series i contributes the lagged copies $X_i(t-\lambda)$ for $\lambda = 0, \ldots, \Lambda$ as columns, and the target's lag-0

Group	Family	Mechanism $f_i(x)$, with $x = X_{\mathbf{pa}_A(i)}$
Homogeneous	Linear	linear with random-sign uniform weights
	Random Fourier features	cosine features; incl. a Gaussian-process variant
	Causal additive	$\sum_j f_{ji}(x_j)$, one random edge function per parent
	Polynomial	a quadratic form with pairwise interactions
	Physical	bounded products, ratios, trigonometric terms and powers
	Multiplicative	product over parent nodes, multiplied by a noise factor
	Sensor array	siblings share a response direction, correlating more than true edges
	Composed	MLPs, trees, piecewise-linear and polynomial maps, randomly aggregated
	Post-nonlinear	invertible post non-linearity after edge functions apply
	Discrete additive	rounded version of the transformed parent plus centered Poisson noise
	Categorical	Dirichlet-drawn conditional probability tables
Node-heterogeneous	Linear, multiplicative, random Fourier features	as in the homogeneous group
	Polynomial	higher powers of each parent
	Interaction	linear plus a few pairwise products of parents
	Rational	a ratio of two linear maps with a positive denominator
	Exponential–logarithmic	exponential or signed logarithm of a linear map
	Periodic	a sum of sinusoids of each parent
	Threshold	a linear map with a single kink
	Saturating	tanh or clipping of a linear map
	Regime mixture	a sigmoid gate on one parent switches between two maps
	Random network	a random one-hidden-layer network
	Activation sum	per-parent activations, including step, modulo, rounding and rank
	Regression tree	a random tree of axis-aligned splits
	Discretization	quantile bins of a linear map
	Aggregation	maximum or log-sum-exp over transformed parents
	Formula node	a noiseless exact equation, feeding measurement readouts
	Semantic child	count, threshold flag, ordinal rule, event count or stock–flow sum

Table 5: **Mechanism families** In a homogeneous SCM, one family defines every mechanism f_i, with parameters redrawn for each node. In a node-heterogeneous SCM, each node draws its own family.

column $X_j(t)$ becomes the response. With n series, the table has $n(\Lambda + 1) - 1$ feature columns. For example, two series with $\Lambda = 2$ and target j give the features $X_i(t)$, $X_i(t-1)$, $X_i(t-2)$, $X_j(t-1)$, $X_j(t-2)$ and the response $X_j(t)$. The lag depth Λ and the row stride are drawn per episode, so effects appear at different lags relative to the sampling interval. The estimator scores every pair of columns, $L = \theta_{\mathrm{DAG}}(D_{\mathrm{lag}})$. As i causes j if some past value of i predicts the present of j, the series-level score takes the strongest evidence over positive lags:

$$S_{ij} = \max_{\lambda \geq 1} L_{(i,\lambda),(j,0)}, \qquad i \neq j. \tag{15}$$

Present-time columns are never read as causes, since a same-time association is symmetric and carries no information about direction. Both directions are supervised at the series level, S_{ij} with A_{ij} and S_{ji} with A_{ji}. Since S_{ij} reads only from the past of i into the present of j, and S_{ji} the reverse, the readout, not the estimator, imposes the direction of time.

Interventional pairs. For a node i with at least one parent, an observational dataset is paired with an interventional one,

$$D \sim P_{\mathcal{M}}, \qquad D^{\mathrm{do}(i)} \sim P_{\mathcal{M}^{\mathrm{do}(i)}}, \qquad A^{\mathrm{do}(i)}_{jk} = A_{jk}\,\mathbf{1}[k \neq i], \tag{16}$$

where $\mathcal{M}^{\mathrm{do}(i)}$ sets $X_i := \tilde{x}_i$ and otherwise shares the mechanisms and exogenous noise of $\mathcal{M}$, and its graph $A^{\mathrm{do}(i)}$ drops the edges into i. With shared noise, only the descendants of i change. Taking $\tilde{x}_i$ as a derangement of the observed values of X_i preserves its marginal, so the intervention shows only through the broken dependence on its parents. In training, each dataset is supervised with its own graph, cut-edge scores are pushed down in $D^{\mathrm{do}(i)}$, and all other scores are held at their observational values.

D Detailed Description of Architecture and Training

In this section, we describe the structure of the model in more detail. Table 6, summarizes the hyperparameters used in training. Note that the model supports a maximum of 2048 rows. For datasets with more than 2048 rows we subsample before running inference.

Feature Encoding. Each feature is normalized independently and processed by the same learned column encoder. For observation n and feature j, the encoder produces $\boldsymbol{h}^n_j = \theta_{\mathrm{encoder}}\big(\widetilde{\mathbf{x}}^n_j\big) \in \mathbb{R}^d$, where $\mathbf{x}^n_j$ denotes the normalized input value. Here, d refers to the transformer model dimension.

Target Encoding. In the *target-aware* mode, a learned embedding $\theta_{\text{target}} : \mathbb{R} \to \mathbb{R}^d$ of the observed target value is broadcast across all feature tokens,

$$\boldsymbol{h}_j^n \leftarrow \boldsymbol{h}_j^n + \theta_{\text{target}}(y^n), \qquad j = 1, \ldots, N_{\text{feat}}, \tag{17}$$

so that each column is encoded jointly with the associated target. Here, y^n refers to the column that was randomly chosen as the target. In *null* mode, we append a single learned vector $\boldsymbol{null} \in \mathbb{R}^{d_{\text{attn}}}$ as an additional feature, for all observations in the dataset. Consequently, the head receives $N_{\text{feat}} + 1$ embeddings of which none is privileged. The two modes share all parameters and differ only in what occupies the target slot. For each observational dataset, we randomly choose between the two modes and train the model to generate both target-aware and symmetric embeddings.

Row-Wise Aggregation. We use alternating observation-feature (row-column) attention, similar to the popular tabular foundation model architectures, resulting in cell-level embeddings $\boldsymbol{C}$ that are aware of the effect of other observations as well as features. We then collapse the observation (row) dimension for each feature using permutation-invariant summary statistics. Let $\boldsymbol{H}_j = \{\boldsymbol{h}_j^1, \ldots, \boldsymbol{h}_j^{N_{\text{obs}}}\}$ denote the per-observation representations for the feature j. We obtain

$$\boldsymbol{t}_j = \theta_{\text{agg}}([\text{mean}(\boldsymbol{H}_j) \parallel \text{std}(\boldsymbol{H}_j)]), \tag{18}$$

where θ_{agg} projects the concatenated feature statistics back to the model dimension and $\parallel$ denotes concatenation. A second set of attention blocks then operates across features, producing one context-aware embedding $\boldsymbol{z}_j \in \mathbb{R}^{d_{\text{attn}}}$ for each feature. This lets each feature representation depend on the joint observational distribution while staying compatible with varying numbers of rows and columns.

Directed Edge-Scoring Head. Note that a symmetric similarity function does not orient an edge as it assigns the same score to $i \to j$ and $j \to i$. Consequently, we use separate source and destination projections and compute the directed edge logit as follows:

$$\boldsymbol{q}_i = W_{\text{src}}\boldsymbol{z}_i, \qquad \boldsymbol{k}_j = W_{\text{dst}}\boldsymbol{z}_j, \qquad L_{ij} = \frac{\boldsymbol{q}_i^\top \boldsymbol{k}_j}{\sqrt{R}} + b_i^{\text{src}} + b_j^{\text{dst}}, \qquad i \neq j, \tag{19}$$

where $\boldsymbol{q}_i, \boldsymbol{k}_j \in \mathbb{R}^R$, R is the bilinear projection dimension; $b_i^{\text{src}} = \boldsymbol{u}^\top \boldsymbol{z}_i$ and $b_j^{\text{dst}} = \boldsymbol{v}^\top \boldsymbol{z}_j$ plus a scalar offset are learned scalar projections of the same embeddings, capturing each feature's marginal ability to act as a cause or as an effect irrespective of the pair. Because W_{src} and W_{dst} are distinct, in general $L_{ij} \neq L_{ji}$, allowing the model to represent edge orientation.

b^{src} and b^{dst} cheaply allow hubs and sinks to have high scores without aligning them individually with every vector they are connected to, by raising the edge scores for that feature along with every other feature.

Threshold Calibrator. For the null-mode graph-recovery results, the directed edge-scoring head produces a logit L_{ij} for each candidate edge $i \to j$. We convert this logit to a probability score $p_{ij} = \sigma(L_{ij})$, where $\sigma(t) = (1 + e^{-t})^{-1}$ is the logistic sigmoid function. A checkpoint-specific, graph-adaptive threshold calibrator then converts these probability scores into a binary graph. The calibrator is fit once on independently generated synthetic SCMs and never on an evaluation dataset.

For each synthetic calibration graph, we compute an *oracle probability threshold*: the threshold on p_{ij} that maximizes recovered-graph F1 against that graph's known ground-truth adjacency matrix, with normalized pairwise SHD used to break ties. This oracle threshold is the ridge-regression target. The regressor receives 17 data-dependent, permutation-invariant features computed from the off-diagonal probability scores: log graph width; the score mean and standard deviation; nine score quantiles; the fractions of scores above 0.50, 0.75, and 0.90; the mean maximum score within an unordered node pair; and the mean absolute difference between the two directions of each pair. Together with an intercept, these form an 18-dimensional design vector. The ridge regressor predicts the logit of the oracle probability threshold, applying the sigmoid and clipping the result to $[0.05, 0.995]$ yielding the threshold for a new graph.

We retain $i \to j$ when p_{ij} exceeds this threshold. When both directions of a pair are retained, we keep only the direction with the larger probability score. The calibrator changes only the binary recovered graph and therefore threshold-dependent metrics such as F1 and SHD. It does not affect the logits L_{ij}, probability scores p_{ij}, or AUROC.

Group	Hyperparameter	Value
Optimization	Optimizer	Muon + AdamW
	Muon momentum	0.95
	AdamW (β_1, β_2)	(0.9, 0.95)
	Weight decay	10^{-4}
	Peak learning rate	2×10^{-4}
	LR schedule	constant with linear warmup
	Warmup steps	4,000
	Training steps	450,000
	Batch size (episodes / gradient step)	6
	Gradient clipping (global norm)	1.0
	EMA decay	0.999
Model	Column embedding dimension	128
	Hidden dimension d_{attn}	256
	Attention heads	4
	Row/column attention blocks	3
	Post-pooling node attention blocks	2
	Feed-forward expansion	$2\times$
	Edge-head bilinear rank R	128
	Max rows	2,048
Losses	$\lambda_{\mathcal{L}_{\text{width}}}$, width invariance loss	0.05
	$\lambda_{\mathcal{L}_{\text{collider}}}$, collider invariance loss	0.05
	$\lambda_{\mathcal{L}_{\text{mediator}}}$, mediator supervision loss	0.10
	$\lambda_{\mathcal{L}_{\text{duplicate}}}$, duplicate parity loss	0.10
	$\lambda_{\mathcal{L}_{\text{conf}}}$, hidden confounder invariance loss	0.05
	$\lambda_{\mathcal{L}_{\text{mech}}}$, mechanism-invariance contrastive loss	0.10
	$\lambda_{\mathcal{L}_{\text{do}}}$, paired intervention loss	0.05

Table 6: ALICE training hyperparameters.

Loss terms. All terms act on the model's matrix of pairwise edge scores. Several auxiliary losses append synthetic columns to a training table. In those cases the model is also trained to recover the edges of the enlarged table, at full weight. The weights listed in the Table 6 applies only to the additional penalty described for that loss. Where a loss compares against the unaugmented table, the reference scores are held fixed, so no gradient flows through them.

Edge loss. Binary cross-entropy over every ordered pair of distinct columns, with true edges up-weighted by the ratio of non-edges to edges in that graph (capped at 50) so that sparse graphs are not dominated by negatives.

Width invariance loss. Appends a block of irrelevant columns (a separately generated causal system plus a few pure-noise columns) with no path to the original variables, and penalizes the squared change in the scores among the original columns. Adding unrelated variables should not change what the model believes about the existing ones.

Mechanism-invariance contrastive loss. For a small batch of graphs, two tables are generated from each graph using different edge functions. The adjacency scores of the tables drawn from the same graph are encouraged to be more similar, while those drawn from different graphs are encouraged to be more different. Edge predictions should depend on the graph, not on the functional form of its mechanisms. Both views also receive the edge loss.

Collider robustness loss. Appends a new column that is a random nonlinear function of two existing variables, making it their common effect. The squared change in the scores between the two parents, in both directions, is penalized: a shared effect creates dependence between its causes but no edge between them.

Mediator supervision loss. Appends a two-step chain, source $\to$ mediator $\to$ outcome, generated from an existing column. Cross-entropy pushes the two direct links up and the source-to-outcome shortcut down, so the model learns that indirect dependence is not a direct edge.

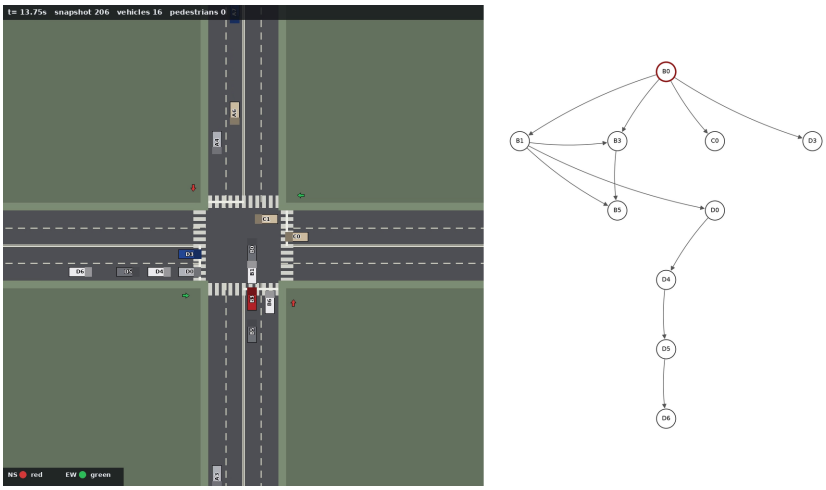


Figure 4: **Example simulation** A visualization of one of the episodes evaluated upon. A car stops suddenly in the intersection and causes a rear-end collision with its follower, and cars in the perpendicular path remain at their stop line since they cannot proceed. The resulting DAG is complex and difficult to reconstruct using statistics alone.

Duplicate parity loss. Appends a near-copy of one of the target's parents, labeled as a parent of the target, and penalizes the squared difference between the original's and the copy's scores toward the target. Interchangeable causes should receive equal credit.

Hidden confounder invariance loss. Appends two columns driven by a common latent variable that is not in the table, with no edges between them or to anything else. The pair is strongly correlated but causally unrelated; the model must score it as having no edge, and the scores among the original columns are penalized for changing.

Paired intervention loss. Pairs an observational table with one generated after a hard intervention on a single variable, which removes that variable's incoming edges. The objective is an equally weighted sum of three parts: the edge loss on both tables; a penalty pushing the scores of the removed edges down in the intervened table; and a smooth-ℓ_1 penalty keeping every other edge score at its observational value.

E Simulating Traffic Incidents with Counterfactual Causal Ground Truth

The motivation of this simulation is to create data that is difficult to infer upon using just an LLM or just a numerical model. We simulate a traffic intersection, and cause events that result in jams. We also store the velocities of all the entities in the frame, and posit the task of reconstructing the causal graph and identifying what chain of events took place.

See Figure 4 for an example, and note how the blame for the stoppage of vehicles D_4, D_5, D_6 is attributed in a *causal chain* to the vehicle directly in front of it. The resulting DAG of such simulations is complex and difficult to reconstruct using velocities alone.

Scene and signal. Each episode is a 15 s simulation of a four-way traffic intersection, integrated at 60 Hz, with two lanes per approach. Vehicles either proceed straight or turn right from the curb lane. They never change lanes. We construct the traffic as a *clean room*. All vehicles share one desired speed. Consequently, in the absence of a disturbance, no vehicle ever decelerates because of another vehicle. The only slowdowns are stops at red lights. For most root types, we choose the incident timing so that the root's own approach displays a green traffic light when the incident occurs. The crossing approach then receives its green later in the episode and finds the junction obstructed. This is the mechanism by which a disturbance on one axis of travel propagates to the other.

Two-world counterfactual and causal ground truth. We simulate every episode twice from the same seed. The two runs share the same arrival schedule and the same signal plan. The *baseline* world contains no incident. The *factual* world is identical except that a single scripted root event occurs. In the factual world, each vehicle's controller computes two accelerations at every step: its *free* acceleration, which it would take in isolation under the signal schedule, and its *realized* acceleration. The controller also identifies the agent responsible for the difference. We integrate this difference online and attribute it to the responsible agent. We add an edge $i \to j$ to the causal graph once the speed deficit that i imposes on j exceeds a threshold, provided that i is itself causally downstream of the root. We exclude any deficit that j's own red light would have imposed regardless, and we assign no label while a vehicle is outside the frame. Finally, we validate the graph against the baseline from both directions. Every vehicle labelled as affected must diverge from its baseline counterpart. Every vehicle that diverges substantially from its counterpart must be labelled as affected.

Root events and the emergency vehicle. We use six root event types. In every type that acts on an existing vehicle, the target is cruising normally near the junction, not already braking for a signal, so that the incident is visually salient.

- *Abrupt brake.* a car brakes hard for a short period and then resumes.
- *Stalled vehicle.* a car comes to rest inside the junction and remains for the rest of the episode.
- *Rear-end.* inattentive follower collides with its leader that comes to rest in the junction
- *Road block.* a car sheds debris behind itself, into the junction.
- *Bus stop dwell.* a bus comes to rest with its body across the junction.
- *Emergency vehicle.* an ambulance enters just as its own light turns red. We choose its entry so that it is never obstructed and no collision occurs.

Datasets. We retain a seed only if it satisfies three conditions: it passes the counterfactual validation, no bodies overlap and no unintended collision occurs, and every labelled event is visible on screen. Each episode provides three outputs: per-vehicle velocity time series, a short textual description of the scene, and the ground-truth causal graph to check performance. We select fifteen such episodes in total.

F Identifiability Analysis

We evaluate whether ALICE can recover causal direction from two sources of bivariate identifiability: non-Gaussianity and nonlinearity. For each dataset, we first independently sample a latent cause and an exogenous noise variable:

$$Z_X \sim \mathcal{N}(0,1), \qquad Z_N \sim \mathcal{N}(0,1), \qquad Z_X \perp Z_N. \tag{20}$$

Both variables are independently transformed using the signed-power function

$$T_q(z) = \operatorname{sign}(z)|z|^q. \tag{21}$$

The observed cause and noise are then given by

$$X = T_q(Z_X). \tag{22}$$

$$N = T_q(Z_N). \tag{23}$$

Although X and N are generated separately and remain statistically independent, their marginal distributions are both controlled by q. When $q = 1$, the transformation is the identity and both variables are Gaussian. Values $q < 1$ compress large magnitudes and produce lighter-tailed non-Gaussian distributions, whereas values $q > 1$ amplify large magnitudes and produce heavier-tailed distributions.

The effect is generated using the additive-noise mechanism

$$Y = X + bX^3 + N, \tag{24}$$

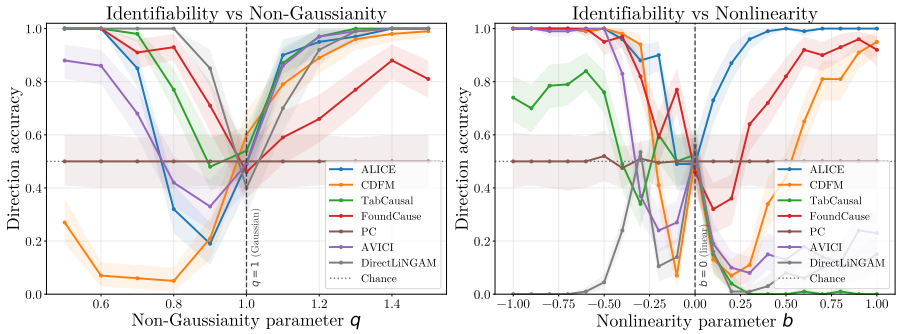


Figure 5: **ALICE identifies causal direction more consistently across non-Gaussian and nonlinear regimes than current CDFMs.** We compare the direction accuracies of various CDFMs on synthetic bivariate data generated using the following relationship: $Y = X + bX^3 + N$. Specifically, we vary the non-Gaussianity q and nonlinearity b. The direction accuracy metric compares the true-edge and reverse-edge logits without thresholding, using 100 balanced datasets per point. The dashed lines mark the non-identifiable linear-Gaussian boundaries.

where b controls the strength and sign of the nonlinear component. We consider two complementary experiments. To isolate identifiability from non-Gaussianity, we set $b = 0$ and vary q from 0.5 to 1.5 in increments of 0.1. To isolate identifiability from nonlinearity, we set $q = 1$ and vary b from -1 to 1 in increments of 0.1. The observed variables X and Y are independently standardized to zero mean and unit variance after generation.

For each parameter value, we generate 50 datasets, balanced equally between the two causal orientations. Direction accuracy is the proportion of datasets for which the true causal direction receives a higher score than its reversal. When $q = 1$ and $b = 0$, the data follow a linear-Gaussian structural causal model, for which causal direction is not identifiable from observational data alone. Accuracy is therefore expected to approach chance at this boundary.

ALICE performs consistently across both identifiable regimes while exhibiting the expected decline near the linear-Gaussian boundary. In comparison, existing causal discovery foundation models often perform well only for particular forms of non-Gaussianity or exhibit strong asymmetries under nonlinear mechanisms.

G Evaluation Metrics

For a dataset with N_{feat} variables, let $A \in \{0,1\}^{d\times d}$ be the ground-truth adjacency, with $A_{ij} = 1$ denoting $X_i \to X_j$. Consistent with the model definition, ALICE returns directed edge logits $L \in \mathbb{R}^{d\times d}$ and probabilities $p_{ij} = \text{sigmoid}(L_{ij})$. We write $P = (p_{ij})$, $\Omega_d = \{(i,j) : i \neq j\}$, and $\mathcal{V}(P) = (p_{ij})_{(i,j)\in\Omega_d}$. Self-edges are excluded from every metric.

Threshold-free Graph Recovery. Our primary graph-recovery metric is directed edge AUROC, computed by comparing $\mathcal{V}(P)$ with $\mathcal{V}(A)$. Equivalently,

$$\text{AUROC}(P, A) = \Pr(p_{e^+} > p_{e^-}) + \tfrac{1}{2}\Pr(p_{e^+} = p_{e^-}), \tag{25}$$

where e^+ and e^- are uniformly sampled true-edge and non-edge slots, respectively. The half credit makes the metric tie-safe. AUROC evaluates the ranking of all ordered candidate edges and does not require a decoding threshold.

For datasets used specifically to test direction, let $\mathcal{U}(A) = \{\{i,j\} : A_{ij} + A_{ji} = 1\}$. Orientation accuracy compares the two directions within each such pair:

$$\text{Orient}(P, A) = \frac{1}{|\mathcal{U}(A)|} \sum_{\{i,j\}\in\mathcal{U}(A)} \left[\mathbf{1}\{(p_{ij} - p_{ji})(A_{ij} - A_{ji}) > 0\} + \tfrac{1}{2}\mathbf{1}\{p_{ij} = p_{ji}\}\right]. \tag{26}$$

Thresholded graph recovery. When a binary graph is required, each method is decoded at its fixed, method-native operating point, yielding $\widehat{A}$. We do not select a threshold from the evaluation labels.

With true positives TP, false positives FP, and false negatives FN over Ω_d, we report

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \qquad \mathrm{Rec} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \qquad F_1 = \frac{2\ \mathrm{Prec}\,\mathrm{Rec}}{\mathrm{Prec}+\mathrm{Rec}}. \tag{27}$$

For the real-world graph benchmarks, structural Hamming distance (SHD) counts one disagreement per unordered node pair:

$$\mathrm{SHD}(\widehat{A}, A) = \sum_{1\le i<j\le d} \mathbf{1}\Big\{(\widehat{A}_{ij}, \widehat{A}_{ji}) \neq (A_{ij}, A_{ji})\Big\}, \tag{28}$$

so a reversal has unit cost.

Paired-view Invariance Evaluation. An invariance episode (Table 7) contains two aligned views $D^{(a)}$ and $D^{(b)}$ of the same causal graph. After restricting both predictions to their shared variables, consistency is the Spearman correlation of the directed edge-score rankings,

$$\mathrm{Cons}(P^{(a)}, P^{(b)}) = \rho_{\mathrm{S}}\Big(\mathcal{V}(P^{(a)}), \mathcal{V}(P^{(b)})\Big). \tag{29}$$

We report consistency together with,

$$\text{Mean view AUROC} = (\tfrac{1}{2}[\mathrm{AUROC}(P^{(a)}, A) + \mathrm{AUROC}(P^{(b)}, A)]) \tag{30}$$

so a model cannot score well merely by being consistently wrong.

Weighted Accuracy. In a bivariate benchmark, each example contains only two variables and the task is to choose their causal direction. For a pair k, let C_k and E_k denote the cause and effect respectively. We compare the two predicted directions using $m_k = p_{C_k E_k} - p_{E_k C_k}$, and assign orientation credit $c_k = 1$ if $m_k > 0$, $c_k = \frac{1}{2}$ for a tie, and $c_k = 0$ otherwise. Tübingen (described in Appendix H.3) additionally provides an official weight w_k for every pair. We therefore report weighted orientation accuracy,

$$\mathrm{Acc}_w = \frac{\sum_k w_k c_k}{\sum_k w_k}. \tag{31}$$

With equal weights, this reduces to ordinary orientation accuracy.

Identifiability sweeps. We evaluate bivariate direction recovery under $Y = X + bX^3 + N$ (as detailed in Appendix F), where b controls nonlinearity and a signed-power parameter q controls the non-Gaussianity of X and N. We sweep q with $b = 0$, and b with $q = 1$. At their intersection, $b = 0, q = 1$, the model is linear-Gaussian and direction is not identifiable from observational data, so ideal accuracy is $\frac{1}{2}$. At every other sweep point it is identifiable from observational data and accuracy should be ideally 1. We report the mean absolute deviation from these ideal accuracies over the sweep (lower is better).

H Evaluation Datasets

We organize evaluation into three complementary classes. **Synthetic datasets** measure graph recovery under controlled variation in graph structure, structural mechanisms, and exogenous noise. **Invariance tests** compare graph-preserving views of the same causal system. **Real-world benchmarks** test transfer to biological, physical, cause-effect, and hydrological systems whose generating processes are not drawn from our training distribution.

H.1 Synthetic Datasets

Synthetic Suite. Inspired from CDFM (Qiao et al., 2026), this suite is a controlled stress test of graph recovery across mechanism families, graph structures, dimensionalities, and sample sizes, with the ground-truth graph known exactly for every dataset. Each dataset is sampled from an SCM. For node j, let X_j denote its observed variable and $\mathrm{pa}_A(j) = \{i : A_{ij} = 1\}$ its parent set under adjacency A. The structural mechanism g_j generates X_j from its parent values and the exogenous noise variable ϵ_j:

$$X_j = g_j\big(X_{\mathrm{pa}_A(j)}, \epsilon_j\big), \tag{32}$$

The suite contains 15 mechanism families: linear, linear with heteroscedastic noise, ordinal linear, post-nonlinear, causal additive, random Fourier feature (RFF), heteroscedastic RFF, ordinal RFF, physics-inspired, heteroscedastic physics-inspired, measurement-error, rounded, discrete additive-noise, conditional-probability-table, and time-lag mechanisms. For continuous mechanisms, the exogenous noise family is sampled from Gaussian, Laplace, clipped Cauchy, standardized Beta, centered exponential, standardized Student-t, standardized lognormal, and Gaussian-mixture distributions. Discrete additive-noise models use centered Poisson noise, while conditional-probability-table models sample categorical conditionals directly.

We sample DAGs from Erdős-Rényi, scale-free, transposed scale-free, Watts-Strogatz, stochastic-block, and geometric random graph families. The evaluation grid uses

$$d \in \{10, 15, 20, 25, 30, 40, 50, 70, 100\}, \qquad N \in \{500, 1000, 2000, 3000, 4000\}, \tag{33}$$

with three independent replicates for every mechanism, dimension, sample-size combination, for $15 \times 9 \times 5 \times 3 = 2{,}025$ datasets. A mechanism family is homogeneous within a dataset, while its parameters, graph family, density, and noise realization vary across datasets. Columns and the corresponding axes of A are jointly permuted before evaluation.

CausalArena. We additionally use the observation-only public split of CausalArena (Li et al., 2026a), restricting evaluation to its *Synthetic* and *Formula* subsets. Each released SCM contributes five independent tables of 1000 observations.

- **Synthetic:** 500 released SCM specifications, hence 2500 tables, over $d \in \{10, 20, 30, 50, 100\}$. The generator varies ten graph families, sixteen mechanism families, fourteen root-distribution families, fifteen noise families, four root-dependence families, and nine difficulty settings. Mechanisms and noise channels may vary locally within one graph.
- **Formula:** 50 released scientific and engineering SCMs, hence 250 tables, with 16-25 variables. Selected nodes implement named equations with declared units and validity ranges. These formula nodes are evaluated exactly from their parents; randomness enters through inputs, measurements, calibration steps, or non-formula readouts, and residual checks verify the stated equations.

Both subsets are scored by directed edge AUROC and F_1.

Intersection-Sim. We design a simulation of a traffic intersection and utilize it for evaluation. Further details are in Appendix E.

H.2 Invariance Datasets

Every invariance episode holds the causal structure fixed while changing one property of the observed view. Tabular views are ordinary $N \times d$ tables and are scored directly after aligning corresponding variables. Temporal views instead contain d jointly observed time series and are evaluated using the lag-table construction and series-level score defined in Appendix C. Thus, ALICE and every other tabular method receive the same lag-expanded inputs, and an ordered pair $i \to j$ is scored only from positive-lag copies of i into the present of j. For the time-series invariance datasets, the lag depth is $\Lambda = 3$. We retain every fourth lagged row and cap each input at 1024 rows.

The complete set of invariance tests, the property changed in each test, and its reported metrics are summarized in Table 7.

Test	Property Change	Metrics
Realization (Tabular)	Two independent tabular realizations from one fixed tabular SCM and DAG.	Mean view AUROC, Consistency
Realization (Time Series)	Two independent time-series realizations from one fixed temporal SCM and DAG.	Mean view AUROC, Consistency
Mechanism (Homogeneous)	One DAG realized with two different mechanism families, each shared by all edges across views.	Mean view AUROC, Consistency
Mechanism (Heterogeneous)	One DAG with edge-specific mechanisms independently redrawn within and across views.	Mean view AUROC, Consistency
Mechanism (Time Series)	One temporal DAG with edge mechanisms, noise, and lags redrawn across views.	Mean view AUROC, Consistency
Width (Noise)	Recovery on original variables after appending 4-12 independent noise columns.	Mean view AUROC, Consistency
Width (Structured)	Recovery of each component before and after adjoining an independent SCM.	Mean view AUROC, Consistency
Identifiability	Direction recovery for $Y = X + bX^3 + \epsilon$ as nonlinearity and non-Gaussianity cross their non-identifiable boundary. (Appendix F)	Ideal-behavior deviation

Table 7: **Invariance Tests.** Consistency is the Spearman rank agreement in Equation (29).

H.3 Real-World Datasets

Causal Chambers. Causal Chambers provides controlled physical systems with experimentally validated causal graphs (Gamella et al., 2025). We use the standard light-tunnel configuration, retaining 20 measured variables and 39 directed edges after removing 18 constant hardware or control variables. The observational environment contains 10000 measurements of light-source controls, electrical current, polarizer controls and angles, and light measurements. We report edge AUROC, F_1, and SHD against the physical reference graph.

Tübingen Cause-effect Pairs. The Tübingen benchmark contains real-world bivariate datasets with annotated causal directions and official pair weights (Mooij et al., 2016). We evaluate the 102 strictly bivariate pairs, excluding six multivariate entries, remove non-finite rows, and retain at most 2,000 observations per pair. The primary metric is the official pair-weighted direction accuracy in Equation (31).

Sachs Protein Signaling. The Sachs dataset contains flow-cytometry measurements of cellular signaling proteins and phospholipids (Sachs et al., 2005). We use the public CD3/CD28 control-condition table with 853 cells, 11 variables, and the 17-edge consensus reference DAG. Each method receives the full table in canonical variable order; we report edge AUROC, F_1, and SHD.

CausalRivers. CausalRivers provides discharge time series together with hydrological upstream-downstream reference graphs (Stein et al., 2025). We use only the eastern-Germany region and the ten benchmark tasks obtained by crossing five subgraph constructions: random, geographically close, root-cause, confounder, and one-plus-random, with graph widths 3 and 5. For each task, a fixed seed selects at most 300 labeled subgraphs. 2724 graphs are evaluated in total across the ten tasks. The discharge records are resampled to six-hour resolution and capped at 4096 time points.

Following the common temporal protocol in Appendix C, ALICE and every other tabular causal discovery method use lag depth $\Lambda = 8$, retain every second lagged row, and receive at most 2048 rows. No tabular method is evaluated on the raw series as if its rows were i.i.d. We average edge AUROC and F_1 within each task and then report the mean over the ten tasks.

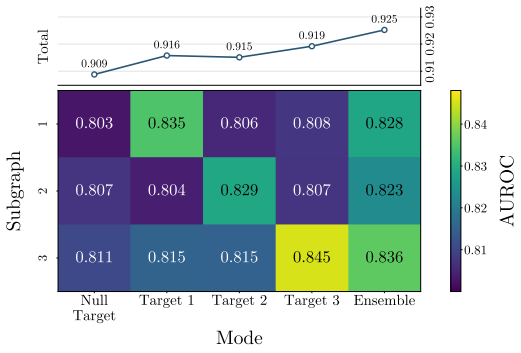
I Extended Results

Invariance	Metric	ALICE (Ours)	FoundCause	CDFM	TabCausal	CauScale	AVICI	PC	DirectLiNGAM
Realization (Tabular)	AUROC	**0.9600**	0.9287	0.8469	0.8552	0.7329	0.7645	0.7916	0.6120
	Consistency	0.8317	**0.8662**	0.8060	0.7551	0.8077	0.8448	0.7327	0.6935
Realization (Time Series)	AUROC	**0.9194**	0.8829	0.8160	0.8067	0.6707	0.7565	0.6021	0.7124
	Consistency	**0.8092**	0.7757	0.6636	0.5989	0.4451	0.6014	0.4879	0.3896
Mechanism (Homogeneous)	AUROC	**0.9337**	0.8489	0.7862	0.7661	0.6860	0.7592	0.6280	0.6251
	Consistency	**0.6245**	0.4818	0.3013	0.2125	0.2376	0.2542	0.2209	0.1140
Mechanism (Heterogeneous)	AUROC	**0.9793**	0.8939	0.8709	0.8658	0.6930	0.8300	0.6363	0.6302
	Consistency	**0.7284**	0.5626	0.4575	0.3910	0.2498	0.3096	0.2789	0.1083
Mechanism (Time Series)	AUROC	**0.8565**	0.7681	0.7351	0.6865	0.7220	0.6581	0.5534	0.6099
	Consistency	**0.4183**	0.2326	0.1914	0.1383	0.2474	0.0780	0.0922	0.0795
Width (Noise)	AUROC	**0.9828**	0.9468	0.9221	0.9035	0.8039	0.8522	0.6746	0.6438
	Consistency	0.8942	0.8381	0.8431	0.5635	0.7729	0.8582	0.6576	**0.9755**
Width (Structured)	AUROC	**0.9835**	0.9524	0.9382	0.9322	0.8157	0.8752	0.6823	0.6564
	Consistency	0.8806	0.8873	0.8135	0.7097	0.7545	0.8056	0.5935	**0.9445**
Identifiability	Non-Gaussianity Deviation	0.1664	0.1618	0.4391	0.0882	0.4282	0.1836	0.4545	**0.0582**
	Non-Linearity Deviation	**0.0595**	0.1643	0.3048	0.6319	0.1629	0.5129	0.4762	0.8650

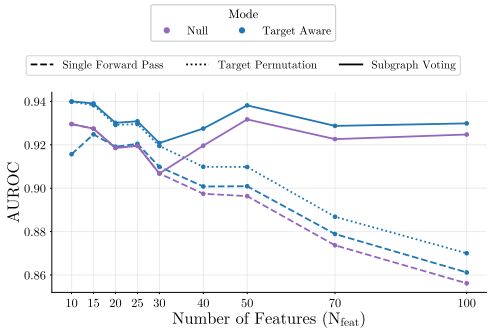
Table 8: **Detailed results for metrics shown in Table 2 across all invariance tests described in Table 7.** ALICE achieves the highest AUROC on every invariance test while maintaining strong consistency across diverse realization, mechanism, and width perturbations, enabled by the invariance losses during pretraining.

		ALICE w/o Invariance Losses		ALICE	
Test	Metric	Null	Ensembled	Null	Ensembled
Realization (Tabular)	AUROC	0.9277	0.9483	0.9511	**0.9630**
	Consistency	0.8668	0.8679	**0.8858**	0.8359
Mechanism (Homogeneous)	AUROC	0.8738	0.9048	0.9157	**0.9429**
	Consistency	0.5164	0.6003	0.6020	**0.6642**
Mechanism (Heterogeneous)	AUROC	0.9539	0.9649	0.9767	**0.9855**
	Consistency	0.6856	0.7230	0.7461	**0.7606**
Width (Noise)	AUROC	0.9781	0.9795	0.9873	**0.9878**
	Consistency	0.8564	0.7899	**0.8897**	0.8220
Width (Structured)	AUROC	0.9746	0.9768	0.9840	**0.9842**
	Consistency	0.8468	0.7911	**0.9068**	0.8029
Identifiability	Non-Gaussianity Deviation	0.2927	–	**0.1655**	–
	Non-Linearity Deviation	0.0914	–	**0.0610**	–
Synthetic Suite	AUROC	0.8772	0.9185	0.8860	**0.9334**
	F1	0.4489	0.5318	0.4866	**0.6004**

Table 9: **Training with invariance losses improves structural robustness and makes target-aware subgraph ensembling more effective.** This ablation compares matched small ALICE models (approximately 2M parameters) trained with and without the invariance losses. We report AUROC and consistency across realization, mechanism, and width-invariance tests, together with identifiability deviations (lower is better) and synthetic-suite performance. For every invariance test, the ensembled result uses target permutation, and for the synthetic suite, it uses target-aware subgraph voting over 64 subgraphs of width 30. Training with invariance losses improves AUROC across all listed tests, generally yields stronger consistency, lower identifiability deviations, and enables larger gains from subgraph voting.



(a) We evaluate target-aware inference of ALICE on graphs constructed from three disjoint local subgraphs. Rows report AUROC within each subgraph, while columns compare null decoding, conditioning on one target from each subgraph, and an ensemble over target-conditioned predictions.



(b) We compare null and target-aware decoding of ALICE under single-pass inference, target permutation, and subgraph voting as the number of features increases. Target-aware subgraph voting achieves the strongest and most stable AUROC, while the non-subgraph modes decline substantially on wider graphs.

Figure 6: **(Left) Target-aware inference improves edge recovery near the selected target, and aggregating complementary target-specific predictions improves the overall graph estimate.** The target-effect experiment shows that conditioning on a target selectively improves recovery in its local subgraph, while ensembling across targets produces the strongest total-graph AUROC. **(Right) Subgraph voting is necessary to retain the target-aware benefit at larger graph widths.** Target-aware subgraph voting preserves strong accuracy as graph width grows, whereas single-pass and target-permutation degrade.

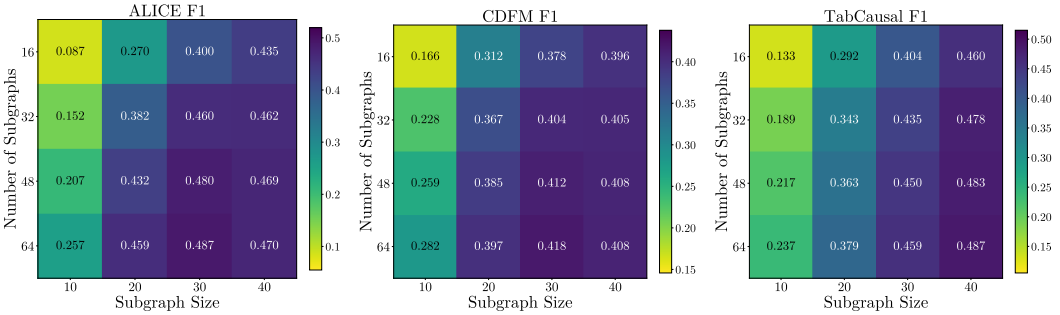


Figure 7: **Subgraph voting generalizes across causal discovery foundation models, with performance improving as more and larger subgraphs are aggregated.** We evaluate CDFM, TabCausal, and ALICE under varying numbers of sampled subgraphs and subgraph sizes on a subset of the synthetic graph suite comprising 30 graphs each at 50, 70, and 100 nodes. Across all three models, F1 rises strongly from small subgraphs and few votes, with the best settings generally using 48–64 sampled subgraphs and subgraph sizes of 30–40.

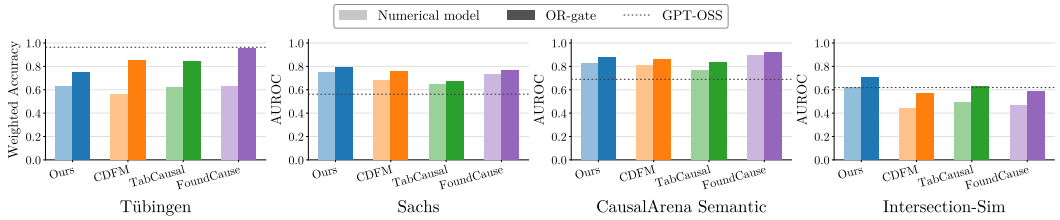


Figure 8: **Feature semantics provide orthogonal information that aids in improving causal estimation.** We use the `GPT-OSS-120B` language model (Agarwal et al., 2025) to understand causal relationships between pairs of features using only semantic context. Consequently, we propose a simple post-processing approach in Section 4 to include this orthogonal information into the predictions from zero-shot causal estimators. Specifically, we observe significant improvements in direction accuracy *(up to 53%)* in the Tübingen benchmark.

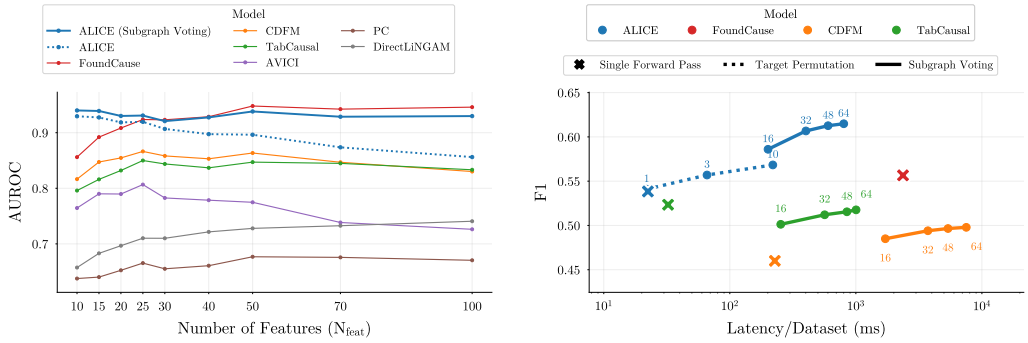


Figure 9: **(Left)**. Subgraph Voting provides significant improvements over the base model for larger number of features. This is accomplished by decomposing the large graph into multiple subgraph inference problems, where the base model is efficient. **(Right) Target-aware subgraph voting provides a favorable accuracy–latency trade-off for ALICE.** We measure end-to-end inference latency on a single NVIDIA H100 GPU and F1 on the synthetic suite for ALICE, FoundCause, CDFM, and TabCausal.

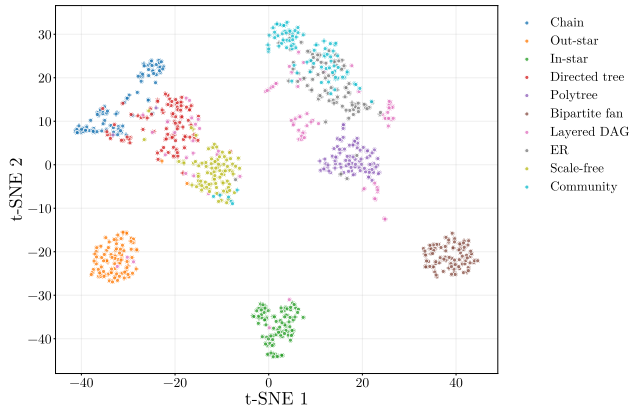


Figure 10: **ALICE's feature embeddings capture graph-family structure, separating synthetic graphs by their underlying topology.** Each point represents a graph-level ALICE embedding obtained from a synthetic graph instance. We project these embeddings into two dimensions with t-SNE and color points by the graph generator family. Graphs from the same family form coherent clusters, while distinct structures, including chains, stars, trees, layered DAGs, and community graphs, occupy largely separated regions, indicating that the learned representation encodes global structural properties of the input graph.