A KNOWLEDGE-INFORMED PRETRAINED MODEL FOR CAUSAL DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Causal discovery has been widely studied, yet many existing methods rely on restrictive assumptions and occupy one of two extremes: they either rely on strong auxiliary supervision, such as costly interventional signals or partial ground-truth, or adopt purely data-driven paradigms without guidance, which hinders practical deployment. Motivated by real-world scenarios where coarse domain knowledge is available, we propose a knowledge-informed pretrained model for causal discovery that integrates weak prior knowledge as a middle ground between the two extremes. Our model adopts a dual-source encoder-decoder architecture to jointly process observational data and structural knowledge while preserving their distinct semantics. We design a diverse pretraining dataset and a curriculum learning strategy that smoothly adapts the model in a knowledge-informed way to varying prior strengths across mechanisms, graph densities, and variable scales. Extensive experiments on in-distribution, out-of-distribution, and real-world datasets demonstrate consistent improvements over existing baselines, while exhibiting robust performance across diverse settings and promising practical applicability.

1 INTRODUCTION

Causal discovery aims to recover underlying dependencies from observational data and is fundamental to explaining how complex systems operate, especially when randomized controlled trials (RCTs) are costly and ethically constrained. It has broad impacts in domains such as industrial process control (Menegozzo et al., 2022), biological networks (Sachs et al., 2005), environmental systems (Stein et al., 2025), and policy evaluation (Athey, 2015), where understanding causal mechanisms is often more valuable than prediction alone. Structural Causal Models (SCMs) (Pearl, 2009) provide a formal framework for causal analysis. An SCM represents causal relationships using a directed acyclic graph (DAG), where nodes denote variables and directed edges represent causal influences. This formalism motivates causal discovery algorithms that estimate a DAG (and associated mechanisms) from observational data, seeking structures consistent with the observed distribution.

Despite steady progress, causal discovery in real-world settings remains difficult due to heterogeneous mechanisms. Existing methods rely on idealized assumptions and strong inductive biases that may fail under mechanism mismatch (Colombo & Maathuis, 2014; Shimizu et al., 2006; Zheng et al., 2018). Some methods try to incorporate prior knowledge through rigid forms that require explicit edge specification (Kalainathan et al., 2020; Zheng et al., 2020). Although these methods can improve performance, this rigidity limits flexibility and applicability. Recent pretrained approaches improve inference efficiency but still have important limitations: they either rely on costly interventional signals (Ke et al., 2022; Lorch et al., 2022) or operate purely data-driven (Dhir et al., 2025), leaving a gap between precise priors and pure observations.

In practice, real-world knowledge is often available in a weaker form, while remaining informative for causal discovery. As illustrated in Fig. 1(A), an expert may distill experience into knowledge that suggest both plausible (dashed lines) and infeasible relations (the red one). Importantly, plausible relations may be direct causal effects, e.g., *Feeding Ratio* $\rightarrow$ *Output Density*, or an indirect one, e.g., *Feeding Ratio* $\rightarrow$ *Reaction Time* in Fig. 1(B). Both direct and indirect relationships provide heuristic guidance in causal discovery. We will provide a more formal definition in Section 3.3.

To fully leverage such knowledge, we propose ***Kode***, a knowledge-informed pretrained model for causal discovery. To the best of our knowledge, Kode is the first pretrained model for causal

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.

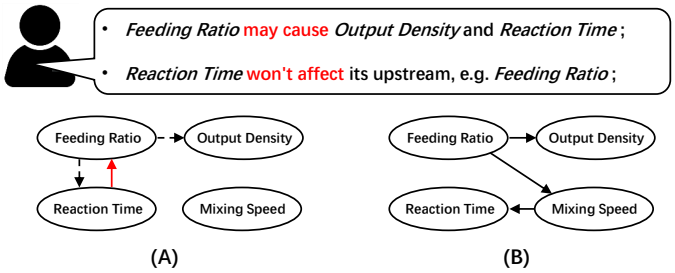


Figure 1: A simple causal structure in a chemical process. (A) Expert knowledge is structured as a graph. (B) The corresponding ground-truth causal graph.

discovery that jointly leverages observational data and prior knowledge. Kode introduces a dual-source encoder-decoder structure that preserves the distinct semantics of data and knowledge while enabling cooperative inference. To train such a model, we build a diverse synthetic training dataset spanning heterogeneous mechanisms, graph densities, and variable scales, together with a curriculum strategy that smoothly schedules prior strength. Extensive evaluations including in-distribution, out-of-distribution, and real-world datasets demonstrate consistent improvements over strong baselines, even when no prior knowledge is provided. In summary, we have four contributions:

- We introduce the first knowledge-informed pretrained model for causal discovery, establishing a new practical paradigm for real-world applications.
- We propose a dual-source encoder-decoder architecture that enables joint inference through semantic interaction between data and prior knowledge.
- We construct a diverse training corpus and implement a curriculum learning strategy to improve adaptation across varying data distributions and prior strengths.
- We provide extensive evaluations demonstrating that our method consistently outperforms baselines across diverse graph sizes, mechanisms, and sparsity levels.

2 Related Work

Within the SCM (Pearl, 2009) framework, a substantial body of work has been developed. Existing approaches can be broadly categorized into three types (Glymour et al., 2019). Constraint-based approaches (Spirtes & Glymour, 1991) infer causal relations via conditional independence tests. Function-based methods achieve identifiability by imposing explicit assumptions on generating mechanisms (Shimizu et al., 2006; Hoyer et al., 2008). Score-based methods (Chickering, 2002) apply a specific score function to guide the search for causal structure. More recently, NOTEARS (Zheng et al., 2018; 2020) introduces a differentiable score for DAG learning and has inspired a series of extensions (Yu et al., 2019; Ng et al., 2019; Lachapelle et al., 2019; Zhu et al., 2019). Despite their empirical success, these methods introduce distinct forms of inductive bias. Classical methods build on specific assumptions, whereas NOTEARS imposes additive equal-variance noise implicitly (Reisach et al., 2021). To improve practical usability, several implementations incorporate prior knowledge. Prior knowledge is either used to reduce the search space (Kalainathan et al., 2020), optimization space (Sun et al., 2023; Mensch, 2023), or as a reward function (Zhu et al., 2019; Wang et al., 2021). These rigid formulations cannot easily represent heuristic or uncertain domain knowledge.

In recent years, transformers have shown promising potential. TabPFN (Hollmann et al., 2025) demonstrates that pretrained tabular models can capture causal information. Meanwhile, attention-based approaches for causal discovery have emerged. These models are pretrained on a synthetic dataset and infer via a single forward pass. Representative methods include AVICI (Lorch et al., 2022), CSIvA (Ke et al., 2022), and BCNP (Dhir et al., 2025), which share similar encoding paradigms but adopt distinct decoder designs with different modeling emphases. However, these methods tend to fall into two extremes: either rely on interventional signals, which are often costly in practice, or follow a purely data-driven paradigm without any prior knowledge. To bridge this gap, we propose a knowledge-informed pretrained model for causal discovery.

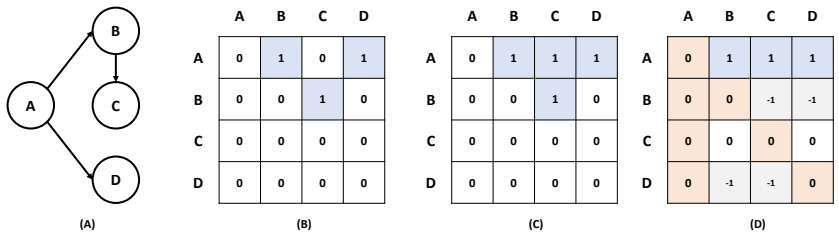


Figure 2: An illustration of knowledge encoding. (A) A symbolic form of Fig. 1. (B) The corresponding adjacency matrix $\mathbf{A}$. (C) The reachability matrix $\mathbf{R}$, which further encodes indirect relations (e.g., $R_{A,C} = 1$). (D) The resulting prior knowledge G_P, where infeasible relations are excluded and self-loops are removed.

3 Preliminary Study

To address these challenges, we first provide formal definitions for causal discovery and knowledge encoding with corresponding motivation.

3.1 Problem of Causal Discovery

Causal discovery is commonly framed as learning a DAG $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ that represents causal relations among variables $\mathbf{X} = \{X_1, \ldots, X_d\}$, where each node corresponds to a variable and directed edges encode causal influences. Under this formulation, causal discovery acts as a posterior estimator that infers the distribution over graph structures given observational data, i.e., $p(\mathbf{G} \mid \mathbf{X})$.

3.2 Motivation

As illustrated in Fig. 1, real-world knowledge is often available in a coarse-grained yet easily attainable form, while remaining informative. Experts can readily provide high-level structural cues, such as procedural order or feasibility constraints, without specifying exact causal mechanisms. We abstract such information as an optional structural-prior matrix G_P. When available, G_P provides auxiliary information for estimating $p(\mathbf{G} \mid \mathbf{X}, G_P)$. This view motivates the development of knowledge-informed models that can flexibly incorporate G_P without over-reliance.

Unlike prior approaches that use partial ground-truth DAGs as hard constraints (Zheng et al., 2020; Kalainathan et al., 2020), we treat prior knowledge as soft, structure-aware signals that guide causal discovery without enforcing exact edges. This formulation avoids treating candidate relations as direct edge labels. In many practical systems, such as Fig. 1, experts may confidently rule out certain relations (e.g., downstream stages affecting upstream ones), identify relations that are plausibly influential, or leave others unspecified due to uncertainty. These three cases can be categorized as *plausible*, *impossible*, and *unknown*. Such coarse yet structure-aware signals preserve the uncertainty inherent in experts' knowledge while still assisting the model in complex, large-scale graphs.

3.3 Knowledge Encoding

Motivated by the three types of relations, we encode prior knowledge structurally. We represent knowledge as a reachability matrix that captures potential causal influence without committing to exact edge existence. Here, we define: $G_P = \mathbf{R} \in \{-1, 0, 1\}^{N \times N}$, where N denotes the number of variables. Each entry R_{ij} represents whether the available knowledge suggests that variable i causally affects variable j. Specifically, we set $R_{ij} = 0$ as *impossible*, and $R_{ij} = 1$ to indicate that i *may* causally influence j, since connectivity within a reachability matrix does not explicitly distinguish between direct and indirect edges. In real-world settings, prior knowledge is often incomplete. Thus, we randomly mask selected entries by replacing them with -1 to denote unknown relations. Since causal graphs are assumed to be DAGs, we further enforce that $R_{ji} = 0$ whenever $R_{ij} = 1$, and set all diagonal entries $R_{ii} = 0$ to prevent self-loops. Unknown masks are applied only to the remaining entries. Notably, this structural representation provides scale-aware guidance even in the absence of positive entries $R_{ij} = 1$, as the variable scale is inherently encoded within the knowledge matrix.

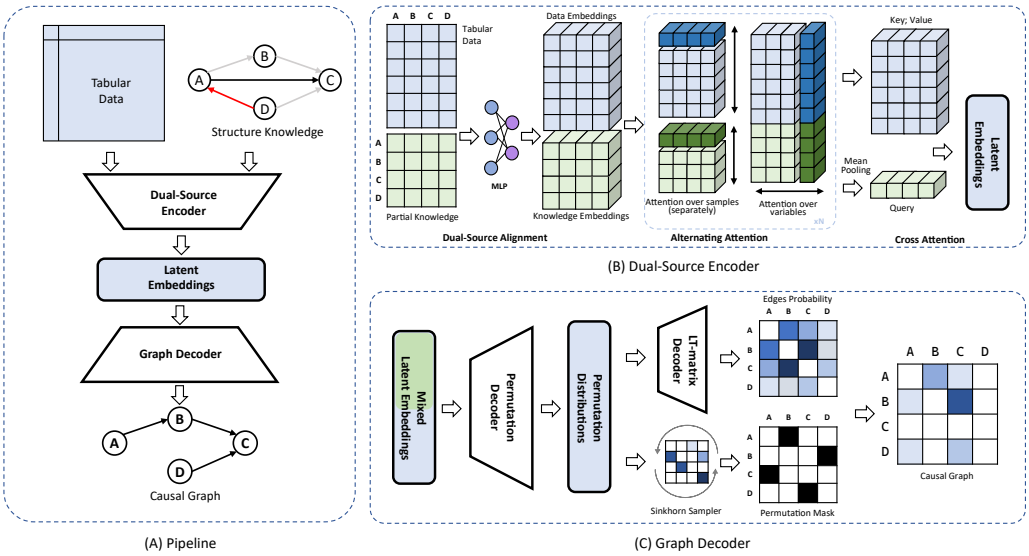


Figure 3: (A) The overall pipeline of Kode, illustrating the global framework that integrates raw data and prior knowledge to infer the causal graph; (B) The dual-source encoder, which achieves alignment and feature compression from hybrid inputs; (C) The graph decoder, responsible for mapping the latent representations back into the final causal structure.

4 Model Architecture

After encoding knowledge structurally, we propose a knowledge-informed architecture with a dual-source encoder and a graph decoder, enabling an end-to-end pipeline that maps observational data and structural priors to causal graphs. An illustration is shown in Fig. 3.

4.1 Dual-Source Encoder

4.1.1 Dual-Source Alignment

With the prior knowledge encoded, we then align observational data and knowledge in a shared latent space. The left panel of Fig. 3(B) illustrates details. We denote the observations by $\mathbf{X} \in \mathbb{R}^{S\times N}$ where S and N are the number of samples and variables. Prior knowledge is encoded into $G_P \in \{-1, 0, 1\}^{N\times N}$ as mentioned previously. Because the two inputs are semantically heterogeneous, we project them into a shared latent space, yielding: $\mathbf{E}_X \in \mathbb{R}^{S\times N\times d}, \mathbf{E}_P \in \mathbb{R}^{N\times N\times d}$ where d denotes the embedding dimension. The two embeddings are then merged into a unified representation, where $S' = N + S$ and $[\cdot||\cdot]$ denotes dimension concatenation:

$$\mathbf{Z} = [\mathbf{E}_P \parallel \mathbf{E}_X] \in \mathbb{R}^{S'\times N\times d} \tag{1}$$

4.1.2 Alternating Attention Mechanism

Following Hollmann et al. (2025), we adopt an alternating attention scheme. Specifically, we design a multi-layer attention module where the attention axis alternates at each layer ℓ. For even-numbered layers, i.e., $\ell \equiv 0 \pmod 2$, the model captures dependencies across samples. Since $\mathbf{Z}$ consists of data and knowledge as Eq. (1), we compute: $Z^{(\ell)} = [\mathrm{Attention}_S(\mathbf{E}_P) \,||\, \mathrm{Attention}_S(\mathbf{E}_X)]$. Importantly, although processed independently, $\mathbf{E}_X$ and $\mathbf{E}_P$ use the same attention module, $\mathrm{Attention}_S(\cdot)$. This parameter sharing encourages the two representations to remain aligned while preserving their distinct semantics. For odd-numbered layers, i.e., $\ell \equiv 1 \pmod 2$, the attention switches to the variable dimension. In this stage, the model focuses on capturing latent dependency structures among variables within each sample. Formally, we compute: $Z^{(\ell)} = \mathrm{Attention}_N\big(Z^{(\ell-1)}\big)$. Unlike sample-wise attention, variable-wise attention operates on the concatenated representation $\mathbf{Z}$ because this stage will not introduce semantic interference. Through this alternation, the model jointly refines by allowing the data and knowledge streams to inform one another. See Fig. 3(B) for illustration.

4.1.3 CROSS ATTENTION MECHANISM

We introduce the cross attention to further integrate representations into a unified latent space. See *Cross Attention* in Fig. 3(B) for illustration. Let L denote the final layer of the alternating attention. We then have: $E_P^{(L)} \in \mathbb{R}^{N\times N\times d}$ and $E_X^{(L)} \in \mathbb{R}^{S\times N\times d}$. We construct the *Queries* $\mathbf{q}$ with mean pooling: $\mathbf{q} = \text{MeanPool}(E_P^{(L)}) \in \mathbb{R}^{N\times 1\times d}$, and let $E_X^{(L)}$ serve as the *Keys* $\mathbf{K}$ and *Values* $\mathbf{V}$ in cross attention, given by: $\mathbf{H} = \text{Attention}(\mathbf{q}, \mathbf{K}, \mathbf{V}) \in \mathbb{R}^{N\times d}$. Functionally, this design constructs a compact representation $\mathbf{H}$, allowing the model to retrieve and aggregate causal information.

4.2 GRAPH DECODER

Following Dhir et al. (2025), we sample DAGs as $\mathbf{G} = \mathbf{Q}\Lambda\mathbf{Q}^\top$, where $\mathbf{Q}$ is a permutation matrix and Λ is a lower-triangular matrix. Before decoding, we augment $\mathbf{H}$ using a weak additive knowledge signal with controlled interaction strength $\epsilon \ll 1$ to accelerate convergence, as $\mathbf{H}' = \mathbf{H} + \epsilon\mathbf{q}$, which allows knowledge-informed inference to softly bias the representation while preserving data-driven signals, thereby guiding inference without disrupting the optimization landscape.

4.2.1 PERMUTATION MATRICES

As illustrated in Fig. 3(C), we project $\mathbf{H}' \in \mathbb{R}^{N\times d}$ into $\mathbf{\Theta} \in \mathbb{R}^{N\times N\times d}$ with an attention-based *Permutation Decoder*. Then we adopt the Gumbel-Sinkhorn relaxation (Mena et al., 2018) with the Hungarian hardening (Charpentier et al., 2022): $\tilde{\mathbf{Q}} \sim \text{Sinkhorn}(\mathbf{\Theta}/\tau + \mathbf{G})$, $\mathbf{Q} = \text{Hungarian}(\tilde{\mathbf{Q}})$, where τ and $\mathbf{G}$ control the temperature and Gumbel noise. We use the straight-through estimator (Bengio et al., 2013) to propagate gradients while keeping discreteness in the forward pass.

4.2.2 LOWER TRIANGLE DECODER

We parameterize edge existence with a lower triangular matrix $\mathbf{\Phi}$, from which Λ is sampled. This naturally enforces acyclicity. To explicitly capture semantic dependencies and maintain representational consistency, we let $\mathbf{Q}$ and Λ share parameters by feeding $\mathbf{\Theta}$ into the lower triangular decoder, as shown in Fig. 3(C), where: $\Phi_{ij} = \mathit{LT\text{-}Decoder}(\mathbf{\Theta})$ if $i > j$, and $\Phi_{ij} = 0$ otherwise. Consequently, the sampled causal graph is deterministically constructed as:

$$\hat{\mathbf{G}} = \mathbf{Q}\,\mathbf{\Phi}\,\mathbf{Q}^\top \tag{2}$$

5 MODEL TRAINING

We train the proposed model using synthetic data paired with structural priors that combines a diverse training dataset with a dedicated training strategy. In addition, we design training objectives that stabilize optimization. Together, they contribute to effective and reliable model training.

5.1 DATA PREPARATION

5.1.1 GENERATION MECHANISM

We construct a large-scale synthetic training dataset, which is diversified along three orthogonal dimensions: *(i) Variable Scale*, *(ii) Graph Sparsity*, and *(iii) Generative Mechanism*. This design exposes the model to a wide range of variations during training and promotes robustness and generalization under complex and realistic scenarios, see 6 Experiments for details. As shown in Tab. 2, the dataset contains 300,000 distinct causal graphs with corresponding observational data. For *(i) Variable Scale*, we consider $N \in \{20, 30, 40\}$; For *(ii) Graph Sparsity*, we adopt the Erdős–Rényi (ER) model (Erdős & Rényi, 1959) with a progressive setting, where the number of edges $|\mathbf{E}|$ ranges over $[20, 60]$, $[30, 90]$, and $[40, 120]$, respectively. As a result, the model is trained over a smooth distributional shift over structural complexity. For *(iii) Generative Mechanism*, we consider linear additive noise models and MLPs for nonlinear models. Specifically, for the linear setting, we define a structural function: $X_i = \sum_{j\in\text{pa}(i)} w_{ij}X_j + \epsilon_i$, where each weight w_{ij} is sampled from a uniform distribution $w_{ij} \sim \mathcal{U}(-10, 10)$, and heteroscedastic noise ϵ_i is sampled as: $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2), \sigma_i \sim \Gamma(\cdot, \cdot)$. In our implementation, we use either $\Gamma(2.5, 2.5)$ or $\Gamma(1, 1)$. This design

differs from the standard homoscedastic-noise assumption (Zheng et al., 2018), thereby providing a more challenging yet realistic training setting. For the nonlinear setting, each variable is generated by a multi-layer perceptron (MLP), taking both its parent variables and an additional latent variable as input. Formally, we define: $X_i = \mathbf{MLP}_i(\mathbf{x}_{\mathrm{pa}(i)}, z_i)$, where the input is formed as $(\mathbf{x}_{\mathrm{pa}(i)} \oplus z_i)$, thereby introducing non-additive stochasticity through concatenation. Each edge is parameterized by an independently initialized three-layer MLP with a hidden width of 32. For each layer of MLP, we define: $Output_l = Layer_l(\phi(Output_{l-1}))$ when $l = 2$, 3 and $Output_l = Layer_l(\mathbf{x}_{\mathrm{pa}(i)} \oplus z_i)$ when $l = 1$, where ϕ denotes the LeakyReLU activation. To balance coverage across graph sizes, densities, and mechanisms, see Tab. 2 for the detailed distribution.

5.1.2 Sampling of Structural Priors

To simulate uncertainty in prior knowledge, we randomly mask the reachability matrix G_P. Concretely, we sample a binary mask $\mathbf{M} \in \{0, 1\}^{N \times N}$ from: $M_{ij} \sim \mathrm{Bernoulli}(\rho)$, with retention rate $\rho \in [0, 1]$. For each entry (i, j), the structured prior knowledge is retained if $M_{ij} = 1$, and set to -1 (unknown) otherwise. During training, we augment reachability-based knowledge with a small fraction (10%) of ground-truth graphs as anchors to guide and stabilize the learning process. We sample with equal probability between $G_P \in \{1, 0, -1\}^{N \times N}$ and $G_P \in \{0, -1\}^{N \times N}$ as different prior strengths, corresponding to different levels of informativeness. To facilitate optimization, unreachable and padding entries are assigned 10^{-10} rather than 0 to preserve weak activity for gradient flow, while diagonal entries are fixed to zero to strictly prohibit self-loops.

5.1.3 Maximum-Size Padding Strategy

To accommodate different graph scales, we adopt zero-padding to use a fixed maximum graph size. In our knowledge-informed setting, zero-padded columns in the observational data $\mathbf{X}$ represent nonexistent variables treated as isolated nodes, while the corresponding rows and columns in the prior matrix G_P encode the absence of both incoming and outgoing relations. Formally, for any $N < N_{\max}$, we have $\tilde{\mathbf{X}} = [\mathbf{X}, \mathbf{0}] \in \mathbb{R}^{S \times N_{\max}}$, and:

$$\tilde{G_P} = \begin{bmatrix} G_P & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \in \{0, 1, -1\}^{N_{\max} \times N_{\max}} \tag{3}$$

Similarly, we replace padded zeros with 10^{-10} to avoid completely inactive representations. To handle dynamic graph scales, several size-agnostic strategies have been proposed (Ke et al., 2022; Lorch et al., 2022; Dhir et al., 2025). However, when data and knowledge share parameters, varying graph sizes may alter attention allocation because the two inputs scale differently and induce a risk of scale-dependent variations in attention allocation (Qin et al., 2022). In contrast, our padding strategy with a small nonzero constant promotes size invariance while enabling more stable optimization. More importantly, when combined with structural knowledge, the padded representations explicitly encode the absence of interactions for non-existent variables, allowing the model to flexibly perceive effective graph size and adapt its causal reasoning across different variable scales.

5.2 Curriculum Learning Strategy

We adopt a curriculum learning strategy (Bengio et al., 2009) to jointly schedule the proportion and strength of sparsity, enabling a stable and progressive training process. From an optimization perspective, curriculum learning organizes training signals so that the model first adapts under mild conditions before gradually facing more restrictive constraints. Such a gradual progression stabilizes optimization and improves convergence by avoiding abrupt changes in the learning landscape.

In the context of the knowledge-informed model, progressive training is particularly important. Sparse knowledge reduces the risk that the model over-relies on prior information, but may provide a weak learning signal early in training, whereas dense priors provide stronger supervision but may overshadow observational data. To mitigate these issues, we adopt a staged training scheme that exposes the model to knowledge with diverse sparsity levels before emphasizing progressively sparser and less informative priors. This progression incrementally bolsters knowledge-informed inference while ensuring robustness through the synergy between data and prior knowledge. As a result, the training process can be characterized by two key factors: the probability of sampling sparse priors and the strength of sparsity constraints. Formally, at epoch e, the sparsity ratio ρ_e is sampled as

$\mathrm{Uniform}(0,1)$ with probability $(1-\pi(e))$ and $\mathrm{Uniform}(0,\rho_{\max}(e))$ with probability $\pi(e)$, where $\pi(e)$ is a monotonically increasing function controlling the probability of sparse sampling. Moreover, the upper bound of sparsity follows a linear annealing schedule: $\rho_{\max}(e) = \max(\tau_0 - \gamma e, \rho_{\min})$, where $\tau_0 = 0.5$ is the initial threshold, $\gamma = 0.02$ is the annealing rate, and $\rho_{\min} = 0.2$ prevents over-constraining the model. Tab. 4 illustrates details. To this end, we implement a curriculum learning strategy in knowledge-informed training, where sparser knowledge is sampled more frequently as training progresses. As a result, the model transitions smoothly toward a sparser and more challenging hypothesis space, thereby mitigating the impact of abrupt distribution shifts.

5.3 Optimization Objective

5.3.1 Similarity Constraint

To enforce structural consistency between the representation and output spaces, we introduce a similarity-based constraint. We suggest that pairwise relationships among samples in the embedding space should be preserved in the outputs. We compute the embedding similarity matrix using cosine similarity, $S_{ij}^{\text{emb}} = \mathrm{sim}_{\cos}(H_i, H_j)$, and define the corresponding output similarity via pairwise Euclidean distances, $S_{ij}^{\text{out}} = 1/(1 + \|\hat{\mathbf{y}}_i - \hat{\mathbf{y}}_j\|_2)$. Both matrices are standardized to remove scale effects. The objective minimizes their discrepancy: $\mathcal{L}_{\text{sim}} = \|S^{\text{emb}} - S^{\text{out}}\|_2^2$. This constraint promotes structure-aware representations by aligning similarities across the two spaces.

5.3.2 Graph Likelihood

Following the paradigm of BCNP (Dhir et al., 2025), we characterize the existence of each edge as an independent Bernoulli random variable according to Eq. (2). By marginalizing over multiple Monte Carlo samples, the training objective is formulated as the minimization of the negative log-likelihood:

$$\mathcal{L}_{\text{graph}} = -\log p(\mathbf{G}) = -\log\left(\frac{1}{K}\sum_{k=1}^{K} p(\mathbf{G} \mid \hat{\mathbf{G}}^{(k)})\right) \tag{4}$$

To enforce the structure-aware optimization, we minimize a combination of the two objectives: $\mathcal{L} = \mathcal{L}_{\text{graph}} + \alpha\mathcal{L}_{\text{sim}}$, where α controls the strength of the similarity regularizer.

6 Experiments

In this section, we introduce the baselines and evaluation metrics, and report experimental results on both synthetic and real-world datasets.

6.1 Setup

We compare our method with representative baselines. **LiNGAM** (Shimizu et al., 2006): a function-based method that assumes particular noise; **PC** (Colombo & Maathuis, 2014): a constraint-based algorithm, searches structure via CI tests; **GES** (Chickering, 2002): a score-based framework that searches for structure greedily; **NOTEARS** (Zheng et al., 2018): a seminal framework for learning a differentiable DAG score; **CSIvA** (Ke et al., 2022): the first pretrained model for causal discovery; **AVICI** (Lorch et al., 2022): improving CSIvA by decoding edge by edge; **BCNP** (Dhir et al., 2025): the first data-driven meta-learner for causal discovery. We evaluate the estimated graphs with two metrics. **SHD**: the minimum edit distance to the true graph. Lower values are better. **F1 score**: the harmonic mean of precision and recall. Higher values are better. See Appendix D for details.

6.2 Synthetic Data

To evaluate our knowledge-informed model, we adopt a standard prior encoded as $\{-1, 0, 1\}$ and a weaker prior containing only negative constraints, encoded as $\{-1, 0\}$. Both are derived from the reachability matrix of the ground-truth. Empirically, most sampled edges represent indirect causal effects (i.e., ancestors rather than parents), thereby mitigating the risk of direct edge supervision. These settings are denoted as *Full Prior* and *Neg Only* in Fig. 4. Edges are randomly masked according to a retention ratio (x-axis), consistent with training. Among these settings, the 0% setting

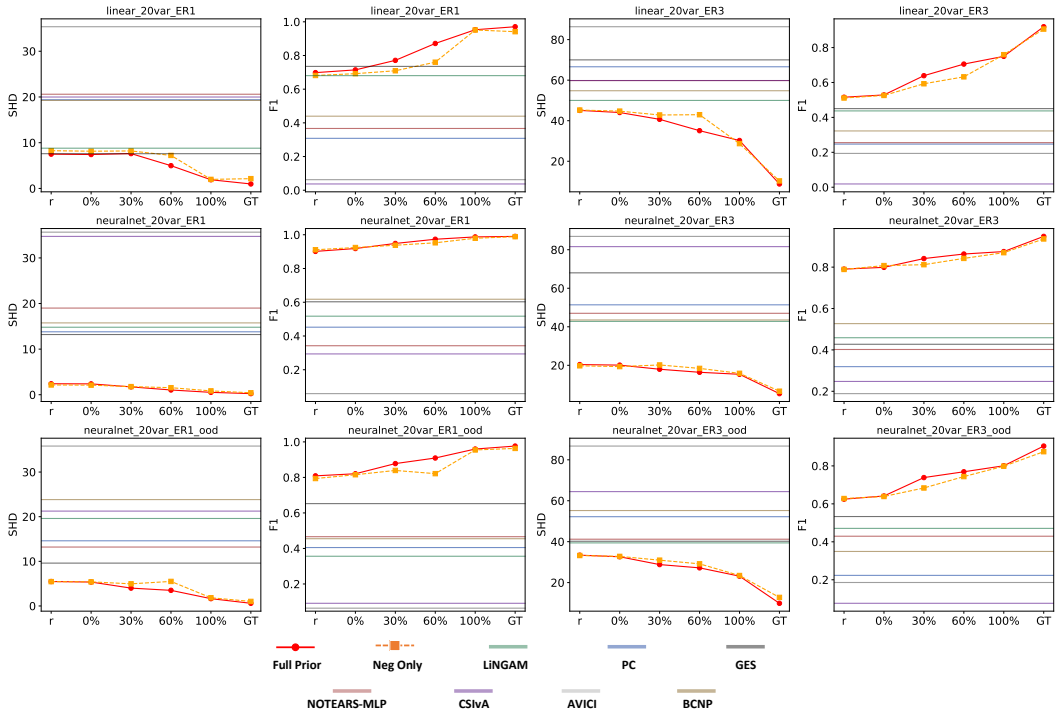


Figure 4: Graph with 20 nodes. The first two rows of subplots evaluate performance on test data following the training distribution. Specifically, in the first row (linear setting), we use heteroscedastic additive noise, which makes causal structure difficult to identify; while in the second row (NN setting), we incorporate non-additive latent variables. The third row, denoted as $\text{NN}_{\text{o.o.d}}$, introduces a severe distribution shift by increasing the network depth to 5 layers, replacing ReLU with sigmoid activations, sampling Latent $\sim \Gamma(2.5,\ 2.5)$ and initializing weights randomly from $\text{W} \sim \mathcal{U}(-50, 50)$.

provides no structural prior and therefore uses only observational data and **GT** represents the special case where the full ground-truth graph is provided as prior knowledge. The leftmost x-axis label, **r**, denotes a random matrix as an incorrect knowledge for robustness evaluation. Performance under varying prior strengths is shown using line plots, with baseline results indicate by horizontal bars. Each subplot reports both SHD and F1 for the same mechanism under varying prior strengths. For each experimental setting, we run 100 independent trials and report the average performance.

We evaluate our model along five dimensions: (i) *Graph Densities*, (ii) *Mechanisms*, (iii) *Out-Of-Distribution (o.o.d.) Generalization*, (iv) *Knowledge Effectiveness*, and (v) *Scales*. For (i)-(iii), we evaluate performance under both linear and nonlinear processes across in-distribution and o.o.d. scenarios, with results for ER(1) and ER(3) graphs reported in Fig. 4. For (iv), the line plots in Fig. 4 illustrate the progression of performance. As the proportion of knowledge increases, the performance of our method improves consistently; meanwhile, gains from negative constraints are predictably less pronounced than from full knowledge. Our method outperforms almost all baselines even without knowledge, showing strong data-driven capability, except that GES excels under linear mechanisms due to its regression-based scoring. Under randomly incorrect priors **r**, performance remains close to the 0% setting, and the model is not overly misled even though the priors contradict the underlying structure. This indicates that, due to our training strategy, the model retains strong data-driven capability without relying excessively on prior knowledge. As a result, it achieves a balance between effectively leveraging correct knowledge and maintaining robustness to empty and incorrect priors. For (v), we illustrate the model's performance across different graph sizes in Fig. 5. As the number of variables increases, our method remains the best-performing method, with only a slight decline in F1 score, even with only 0% knowledge. We hypothesize that this robustness partly arises from the presence and dimensions of the prior matrix: even when it contains no informative relations, its shape still reveals the effective graph size. Overall, our model effectively leverages knowledge to achieve robust inference and strong generalization.

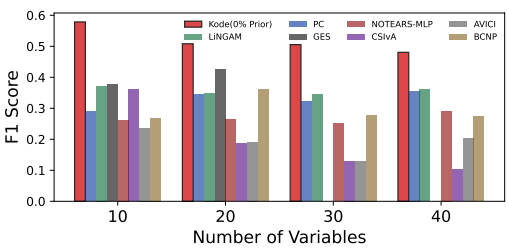


Figure 5: Performance on Gaussian Process ER(3) across scales: ***Kode*** maintains a clear advantage; GES (4^{th} bar) collapses owing to excessive density at $N = 30$ and 40, so its results are omitted.

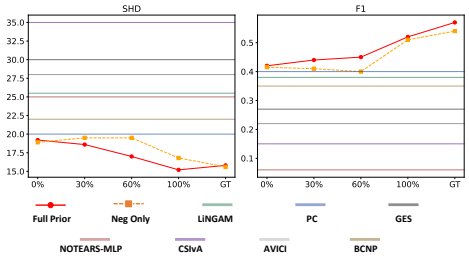


Figure 6: Sachs Experiment. We evaluate two encodings of prior knowledge: $\{-1, 0, 1\}$ and $\{-1, 0\}$. Plots show performance under varying levels of missing knowledge.

Table 1: Industry Experiments. We derive knowledge encoded as $\{-1, 0\}$ from the industrial pipeline topology, capturing the constraint that downstream components cannot be ancestors of upstream ones.

		Kode (topo-priors)	**Kode** (no priors)	**LiNGAM**	**PC**	**GES**	**CSIvA**	**AVICI**	**BCNP**
Food	**SHD**	***80.1***	88.35	89.66	95.23	97.24	105.3	87.8	92.38
	F1	***0.40***	0.35	0.08	0.19	0.32	0.17	0.22	0.28
TEP	**SHD**	***31.26***	35.47	48.82	68.57	124.35	49.77	52.61	35.58
	F1	***0.44***	0.38	0.14	0.14	0.09	0.22	0.28	0.31

6.3 Real-world Data

To evaluate applicability in complex settings, we further test our method on real-world datasets that deviate substantially from the training distribution in terms of mechanisms, scale, and sparsity. We adopt CIPCaD-Bench (Menegozzo et al., 2022) for causal discovery in real-world industrial systems. Within this benchmark, **Ultra-processed Food** models causality in a food production pipeline, containing 17 variables and 83 edges, and can be approximately regarded as a dense ER(5) graph. This is particularly challenging as the training data reaches a maximum density of only ER(3). **Tennessee Eastman Process** originates from a chemical process system and consists of 33 variables with 32 edges, corresponding to a sparse ER(1) graph. We illustrate them as *(with topo-priors)* in Tab. 1. In addition, we include the **Sachs** dataset (Sachs et al., 2005), a widely used benchmark in biological causal discovery. This dataset captures protein signals and contains 11 variables with 18 edges, which can be viewed as an ER(1.5) graph. This is also challenging as the model has never encountered graphs at such a small scale during training. Furthermore, all three datasets feature complex, out-of-distribution mechanisms beyond the training ones.

As shown in Fig. 6 and Tab. 1, we evaluate all methods on three real-world datasets. The best and second-best results are highlighted in bold and underlined, respectively. These benchmarks involve substantial shifts in the mechanism, density, and graph size relative to the training data. Therefore, compared to synthetic data, the absolute performance decreases. Nevertheless, the model remains competitive and outperforms baselines. This demonstrates the model's practical utility in realistic scenarios. The proposed model benefits from diverse pretraining and the synergy between mixed-scale training and prior knowledge for robust, scale-aware reasoning. We believe the performance gap can be further narrowed with deeper architectures and more comprehensive training data.

7 Conclusion

In this work, we propose the first knowledge-informed pretrained model for causal discovery, introducing a new paradigm that integrates data and coarse-grained knowledge. We develop a tailored model architecture, with a corresponding training dataset and curriculum learning strategy to support the model. Extensive experiments demonstrate its consistent superiority across diverse settings. Together, these results highlight the effectiveness and generality of our approach for causal discovery under varying mechanisms, scales, and levels of prior knowledge, in both i.d. and o.o.d. settings.

AI USE STATEMENT

Generative AI tools were used solely to improve the readability, grammar, and linguistic presentation of author-written text. They were not used to formulate the research questions or hypotheses, develop the methodology, implement the proposed method, generate or process data, design or conduct experiments, analyze or interpret results, formulate mathematical claims, construct proofs, or produce the scientific conclusions of this work. The authors reviewed and verified all AI-assisted edits and take full responsibility for the accuracy, originality, and final content of the paper.

REPRODUCIBILITY STATEMENT

The model architecture, structural-prior encoding, training objectives, and curriculum schedule are described in the main paper. The appendix provides the synthetic graph and data-generation procedures, prior sampling distributions, baseline implementations, hyperparameters, model configuration, computational hardware, evaluation metrics, and additional results. Each main synthetic evaluation setting comprises 100 independent trials.

ETHICS STATEMENT

This work does not involve human-subject experiments or the collection of personally identifiable or sensitive information. We use publicly available datasets in accordance with their terms of use. The proposed method does not introduce risks beyond those generally associated with machine-learning systems, although its predictions may inherit biases or errors from the training data. Deployment in safety-critical or high-stakes settings should therefore include domain-specific validation and appropriate human oversight. We have considered research integrity, privacy, fairness, and potential misuse, and are not aware of any violation of the ICLR Code of Ethics.

REFERENCES

Susan Athey. Machine learning and causal inference for policy evaluation. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 5–6, 2015.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Bertrand Charpentier, Simon Kibler, and Stephan Günnemann. Differentiable DAG sampling. In *International Conference on Learning Representations*, 2022.

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

Diego Colombo and Marloes H Maathuis. Order-independent constraint-based causal structure learning. *J. Mach. Learn. Res.*, 15, 2014.

Anish Dhir, Matthew Ashman, James Requeima, and Mark van der Wilk. A meta-learning approach to bayesian causal discovery. In *International Conference on Learning Representations*, volume 2025, pp. 14158–14178, 2025.

Paul Erdős and Alfréd Rényi. On random graphs i. *Publ. math. debrecen*, 6:290–297, 1959.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

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.

Diviyan Kalainathan, Olivier Goudet, and Ritik Dutta. Causal discovery toolbox: Uncovering causal relationships in python. *Journal of Machine Learning Research*, 21(37):1–5, 2020.

Nan Rosemary Ke, Silvia Chiappa, Jane X Wang, Jorg Bornschein, Anirudh Goyal, Melanie Rey, Matthew Botvinick, Theophane Weber, Michael Curtis Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. In *ICML 2022: Workshop on Spurious Correlations, Invariance and Stability*, 2022.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. In *NeurIPS 2019 Workshop on Solving Inverse Problems with Deep Networks*, 2019.

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.

Gonzalo Mena, David Belanger, Scott Linderman, and Jasper Snoek. Learning latent permutations with gumbel-sinkhorn networks. In *International Conference on Learning Representations*, 2018.

Giovanni Menegozzo, Diego Dall'Alba, and Paolo Fiorini. Cipcad-bench: Continuous industrial process datasets for benchmarking causal discovery methods. In *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, pp. 2124–2131. IEEE, 2022.

Arthur Mensch. Lbfgs wrapper for pytorch. `https://gist.github.com/arthurmensch/c55ac413868550f89225a0b9212aa4cd`, 2023. GitHub Gist.

Ignavier Ng, Shengyu Zhu, Zhitang Chen, and Zhuangyan Fang. A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*, 2019.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Zhen Qin, Xiaodong Han, Weixuan Sun, Dongxu Li, Lingpeng Kong, Nick Barnes, and Yiran Zhong. The devil in linear transformer. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2022.

Alexander Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! causal discovery benchmarks may be easy to game. *Advances in Neural Information Processing Systems*, 34:27772–27784, 2021.

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.

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

Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1):62–72, 1991.

Gideon Stein, Maha Shadaydeh, Jan Blunk, Niklas Penzel, and Joachim Denzler. Causalrivers – scaling up benchmarking of causal discovery for real-world time-series. 2025. doi: 10.48550/arXiv.2503.17452.

Xiangyu Sun, Oliver Schulte, Guiliang Liu, and Pascal Poupart. Nts-notears: Learning nonparametric dbns with prior knowledge. In *AISTATS*, 2023.

Xiaoqiang Wang, Yali Du, Shengyu Zhu, Liangjun Ke, Zhitang Chen, Jianye Hao, and Jun Wang. Ordering-based causal discovery with reinforcement learning. *arXiv preprint arXiv:2105.06631*, 2021.

Yue Yu, Jie Chen, Tian Gao, and Mo Yu. Dag-gnn: Dag structure learning with graph neural networks. In *International conference on machine learning*, pp. 7154–7163. PMLR, 2019.

Keli Zhang, Shengyu Zhu, Marcus Kalander, Ignavier Ng, Junjian Ye, Zhitang Chen, and Lujia Pan. gcastle: A python toolbox for causal discovery. *arXiv preprint arXiv:2111.15155*, 2021.

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*, pp. 3414–3425. Pmlr, 2020.

Shengyu Zhu, Ignavier Ng, and Zhitang Chen. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477*, 2019.

Appendix

A Baseline Details

To evaluate the effectiveness of our proposed framework, we compare it against a comprehensive set of representative baselines spanning different causal discovery paradigms.

A.1 Traditional Methods

All traditional and gradient-based baselines are implemented using the open-source library `gcastle` (Zhang et al., 2021).

- **LiNGAM** (Linear Non-Gaussian Acyclic Model) (Shimizu et al., 2006): A functional-based method that leverages non-Gaussianity in the data to identify the unique causal ordering. It assumes that the underlying structural equations are linear and the exogenous noise follows a non-Gaussian distribution, enabling the identification of the full DAG rather than just the Markov Equivalence Class (MEC).
- **PC** (Peter-Clark Algorithm) (Colombo & Maathuis, 2014): The quintessential constraint-based method. It starts with a complete undirected graph and performs a series of conditional independence (CI) tests to remove edges. We utilize the Fisher-z test as the CI engine with a significance level of $\alpha = 0.05$ to determine the skeleton and orientations.
- **GES** (Greedy Equivalence Search) (Chickering, 2002): A two-phase score-based algorithm that searches the space of MECs. It starts from an empty graph, greedily adds edges to increase the Bayesian Information Criterion (BIC) score, and subsequently removes redundant edges. As noted in the main text, GES is constrained by a time threshold of 1000 seconds in our large-scale experiments.
- **NOTEARS-MLP** (Zheng et al., 2018; 2020): A pioneering continuous optimization framework that transforms the discrete DAG search problem into a differentiable one. By utilizing a Multi-Layer Perceptron (MLP) and a matrix-based acyclicity constraint $h(W) = 0$, it extends the NOTEARS framework to handle nonlinear causal relationships.

A.2 Pretrained Methods

A.2.1 Methods

To assess the zero-shot generalization capabilities of our framework, we compare it against state-of-the-art pretrained causal discovery models. All such models are evaluated using the unified implementation provided by the BCNP framework (Dhir et al., 2025), which leverages a shared encoder architecture to standardize the treatment of heterogeneous causal data. They developed corresponding decoders tailored to the specific narratives and theoretical motivations of their respective works.

- **CSIvA** (Causal Structure Identification via Attention) (Ke et al., 2022): This represents the pioneering effort in treating causal discovery as a sequence-to-sequence translation task. It employs a Transformer-based architecture where the encoder processes observational data and the decoder generates a linearized representation of the adjacency matrix. By training on a diverse corpus of synthetic graphs, CSIvA learns to identify causal orderings through self-attention mechanisms.
- **AVICI** (Amortized Variational Inference for Causal Discovery) (Lorch et al., 2022): Building upon the sequence-based foundation of CSIvA, AVICI improves the decoding process by predicting causal edges independently (edge-by-edge) or in blocks. It treats causal discovery as an amortized inference problem, where a neural network learns to approximate the posterior distribution of the graph $P(G|\mathcal{D})$. This approach significantly enhances scalability and robustness when dealing with varying noise distributions and functional forms. Critically, both CSIvA and AVICI rely heavily on interventional signals to resolve causal ambiguities.
- **BCNP** (Bayesian Causal Network Pretraining) (Dhir et al., 2025): As the first fully data-driven Bayesian neural meta-learner for causal discovery, BCNP advances the field by

enabling causal structure learning without requiring interventional data. It frames the discovery process as a meta-learning task, where the model is trained to output a belief over the causal graph. By unifying the encoders across CSIvA and AVICI, the BCNP framework allows for a consistent comparison of how different decoding strategies affect the inference of complex causal topologies.

A.2.2 Training Dataset

Bayesian meta-learning frameworks incorporate diverse functional mechanisms and varying graph densities to enhance robustness, they are inherently limited by their inability to accommodate varying variable scales within a single model.

To achieve competitive performance, the authors of these baselines follow an extensive data generation protocol as outlined in the BCNP framework (Dhir et al., 2025). They construct a large-scale meta-training corpus consisting of 500,000 synthetic datasets, where each dataset contains $N = 1000$ samples but is constrained to a fixed dimensionality of $d = 20$ nodes. Following the recommendations of Reisach et al. (2021), all variables are normalized post-generation to prevent the models from exploiting marginal variance artifacts. The underlying causal DAGs G are sampled from the Erdos-Renyi (ER) distribution with average degrees set to $\{20, 40, 60\}$, representing a wide range of structural densities.

To simulate complex and unknown real-world generation processes, these frameworks employ three distinct functional forms during pretraining:

Linear Mechanisms For linear datasets, each variable X_d is generated as a weighted linear combination of its parents $\text{PA}_G(d)$ plus additive Gaussian noise:

$$(X_d)_i \sim \mathcal{N}(\mathbf{w}^\top (X_{\text{PA}_G(d)})_i, \sigma_i) \tag{5}$$

where the coefficients $\mathbf{w} \sim \mathcal{N}(0, 10)$ determine the causal strength, and the noise variance is sampled from $\sigma_i \sim \text{Gamma}(2.5, 2.5)$. This broad parameter sampling ensures high diversity in linear dependencies across the training tasks.

Neural Network (NeuralNet) Mechanisms To introduce complex non-parametric dependencies, structural equations are modeled using randomly initialized Multi-Layer Perceptrons (MLPs):

$$(X_d)_i = \text{NN}_\theta((X_{\text{PA}_G(d)})_i, \epsilon_i), \quad \epsilon_i \sim \mathcal{N}(0, 1) \tag{6}$$

The architecture employed consists of a two-layer network with a hidden width of 32 and Leaky ReLU activations. For root nodes, the exogenous noise ϵ is passed directly through the network to generate the observed values.

Gaussian Process (GPCDE) Mechanisms To generate highly flexible and continuous functional mappings, these methods utilize Gaussian Processes with an exponential Gamma kernel.

The kernel K is defined as:

$$K(\mathbf{u}, \mathbf{v}) = \exp\left(-\frac{1}{\boldsymbol{\lambda}}\|\mathbf{u} - \mathbf{v}\|^\gamma\right) \tag{7}$$

where the smoothness parameter $\gamma \sim \text{Uniform}(0.1, 1)$ and lengthscales $\lambda \sim \log\mathcal{N}(-1, 1)$ are sampled independently for each input dimension. Given the parent samples and scalar noise $\epsilon \sim \mathcal{N}(0, 1)$, the variable is sampled as:

$$X_d \sim \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma^2\mathbf{I}), \quad \sigma \sim \text{Gamma}(1, 10) \tag{8}$$

By sampling both γ and λ, these frameworks ensure a broad coverage of functions with varying degrees of smoothness and sensitivity, which serves as a rigorous pretraining regime to improve out-of-distribution generalization.

B Training Details

B.1 Graph Generation

To address the issue of graph density in data generation, we first briefly introduce the Erdős–R'enyi (ER) model (Erdős & Rényi, 1959). In the directed Erdős–Rényi graph $\vec{G}(d, p)$, there are d vertices

Table 2: Training dataset configurations under different generative mechanisms.

Variables	Edges	Linear		Neural Networks		
		noise $\sim \mathcal{N}(0,\, d^2)$ $d \sim \Gamma(2.5,\, 2.5)$	$d \sim \Gamma(1,\, 1)$	Latent $\sim \mathcal{U}(-1,\, 1)$	Latent $\sim \Gamma(1,\, 1)$	Latent $\sim N(0,\, d^2)$ $d \sim \Gamma(1,\, 1)$
20	$\lvert\mathbf{E}\rvert \in [20,\, 60]$	20,000	30,000	16,667	16,667	16,667
30	$\lvert\mathbf{E}\rvert \in [30,\, 90]$	20,000	30,000	16,667	16,667	16,667
40	$\lvert\mathbf{E}\rvert \in [40,\, 120]$	20,000	30,000	16,667	16,667	16,667

Table 3: Prior probabilities. Reachability matrices dominate sampling, while a small fraction of ground-truth graphs remains essential.

Prior Type		Probability
Reachability Matrix	(0, 1, -1)	0.45
	(0, -1)	0.45
Ground Truth	(0, 1, -1)	0.05
	(0, -1)	0.05

Table 4: Illustration of how the sampling proportions and sparsity threshold evolve over epochs.

Epochs	Proportion	Sparse Threshold
[0, 5)	No Sparse	/
[5, 8)	8 : 2	(0.34, 0.40]
[8, 10)	6 : 4	(0.30, 0.34]
[10, 15)	4 : 6	(0.20, 0.30]
[15, 50]	2 : 8	0.20

and each of the $d(d-1)$ ordered pairs of distinct vertices is included independently with probability p, so the expected number of directed edges is:

$$\mathbb{E}\big[|E_{\to}|\big] \;=\; p\, d(d-1). \tag{9}$$

Each vertex v has indegree and outdegree distributed as $\mathrm{Bin}(d-1, p)$ and total degree $\deg(v) = \deg^{\mathrm{in}}(v) + \deg^{\mathrm{out}}(v)$ has expectation $2p(d-1)$. Moreover, since both the average degrees satisfy $\frac{|E_{\to}|}{d} = p(d-1)$, and the average total degree is $\frac{2|E_{\to}|}{d} = 2p(d-1)$.

B.2 Data and Knowledge Preparation

Tab. 2 summarizes the dataset, which contains 300,000 distinct causal graphs with corresponding observational data. Samples are adjusted for balancing graph sizes, densities, and mechanisms.

Tab. 3 provides the prior distribution in training dataset. It illustrates reachable matrix and ground truth graphs, including both $G_P \in \{1, 0, -1\}^{N\times N}$ and $G_P \in \{0, -1\}^{N\times N}$.

Tab. 4 illustrates how the sampling proportions and the sparsity threshold evolve over training epochs. Formally, at epoch e, the sparsity ratio ρ_e is sampled from a mixture distribution:

$$\rho_e \sim \begin{cases} \mathrm{Uniform}(0, 1), & \text{with probability } 1 - \pi(e), \\ \mathrm{Uniform}(0, \rho_{\max}(e)), & \text{with probability } \pi(e), \end{cases} \tag{10}$$

where $\pi(e)$ is a monotonically increasing function controlling the probability of sparse sampling. Moreover, the upper bound of sparsity follows a linear annealing schedule: $\rho_{\max}(e) = \max(\tau_0 - \gamma e, \rho_{\min})$, where $\tau_0 = 0.5$ is the initial threshold, $\gamma = 0.02$ is the annealing rate, and $\rho_{\min} = 0.2$ prevents over-constraining the model. Evolving begins at the fifth epoch to ensure general initialization.

B.3 Environments and Model Details

We train all models with AdamW using an initial learning rate of 3×10^{-4}, a learning-rate decay factor of 0.9, a warm-up ratio of 0.1, and a weight decay of 10^{-5}. Training is performed in bfloat16 precision with 8 data-loading workers. All experiments are conducted on synthetic datasets with 20,30,40 variables as mentioned above, where the sample size is randomly varied from 100 to 1000. Our model uses 4 encoder layers and 4 decoder layers, with a hidden dimension of 512, a feedforward dimension of 1024, and 8 attention heads. We perform 1000 Sinkhorn iterations for graph decoder.

Training was conducted on a machine equipped with 8 NVIDIA GeForce RTX 5090 GPUs, each with 32 GB of memory. The model contains 24,396,306 parameters in total.

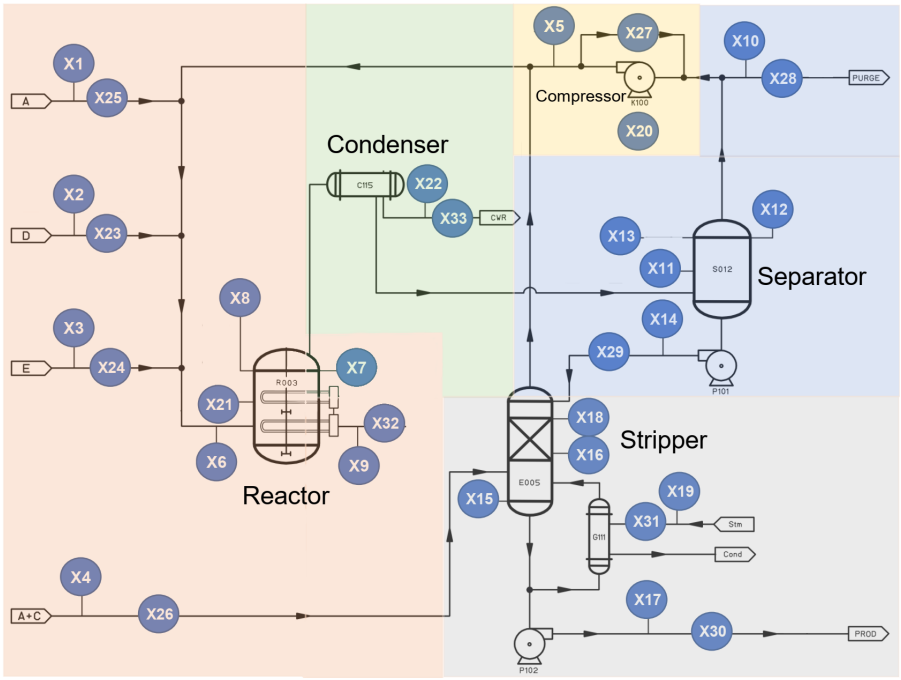


Figure 7: Physical process topology and production stages of the Tennessee Eastman Process.

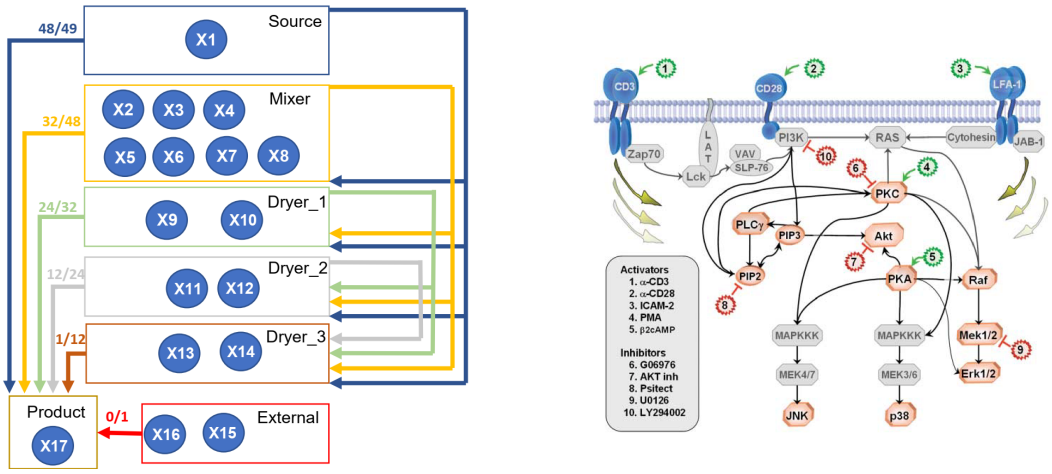


Figure 8: Physical topology of the Food (left) and biological interaction of the Sachs (right).

C Real-world Data Details

To validate our model's performance in real-world industrial scenarios, we evaluate it on two benchmark datasets from the CIPCaD-Bench (Menegozzo et al., 2022), which focus on continuous industrial process dynamics.

- **Tennessee Eastman Process (TEP)**: The TEP dataset is a widely recognized benchmark in the field of industrial process control and fault diagnosis. It originates from a simulated chemical process involving five major unit operations: a reactor, a condenser, a compressor, a separator, and a stripper. Unlike synthetic data, TEP features complex closed-loop control structures, high-dimensional variables, and inherent time delays. We utilize the ground-truth

process topology provided by CIPCaD-Bench, which maps the physical production stages and chemical reaction logic to evaluate the accuracy of causal structure identification in a highly coupled system.

- **The Ultra-Processed Food (Food)**: The Ultra-Processed Food dataset models a real-world food production pipeline, where variables correspond to process parameters and quality measurements across multiple manufacturing stages. It consists of 17 continuous variables with 83 causal edges, forming a highly dense causal graph that can be approximately characterized by an ER(5) model. Due to its multi-stage structure and strong inter-dependencies, Food exhibits complex causal mechanisms and pronounced indirect effects, making it a challenging benchmark for causal discovery methods, particularly in dense and highly coupled industrial settings.

In addition, we include the Sachs dataset (Sachs et al., 2005), a widely used benchmark in biological causal discovery.

- **Sachs**: This dataset captures protein signaling interactions measured from flow cytometry experiments under various experimental conditions. It consists of 11 variables and 18 directed causal edges, forming a moderately sparse causal graph that can be roughly characterized as ER(1.5). Owing to its well-studied ground-truth structure and nonlinear dependencies, the Sachs dataset serves as a standard testbed for evaluating causal discovery methods under biological mechanisms, and complements the industrial datasets by providing a distinct domain with different causal semantics and noise characteristics.

D Additional Experiments

D.1 Metrics

- **Structural Hamming Distance (SHD)** $\downarrow$: SHD quantifies the total edge modifications (additions, omissions, or reversals) needed to match the ground truth. A lower value reflects higher topological fidelity, with zero indicating perfect recovery.
- **F1-Score** $\uparrow$: The F1-score is the harmonic mean of Precision and Recall, balancing the model's ability to identify true causal edges while minimizing spurious ones:

$$\mathrm{F1} = \frac{2 \cdot \mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}. \tag{11}$$

  It ranges from 0 to 1, where higher values signify a superior trade-off between edge detection accuracy and completeness.

D.2 Expanded Results of Main Text

We expanded our evaluation to include more comprehensive experiments across variable scales of 10 (Fig. 12), 20 (Fig. 13), 30 (Fig. 14), and 40 (Fig. 15). Beyond the settings discussed in the main text, we incorporated additional experimental designs to further validate the robustness of our model. Specifically, we evaluate the performance under the following distinct data generation regimes:

- **Linear Mechanisms**: For linear structural equations, we introduced two new types of additive noise:
  - **Normal Noise**: $\epsilon_i \sim \mathcal{N}(0, 1)$, representing the standard parametric assumption.
  - **Uniform Noise**: $\epsilon_i \sim \mathcal{U}(-5, 5)$, in order to evaluate the model's ability to higher-order statistical moments.
- **Nonlinear Mechanisms via Gaussian Processes (GP)**: To evaluate the model under complex, unknown functional forms, we employ GP priors to generate nonlinear causal mechanisms. For each variable X_i, the structural equation is formulated as:

$$X_i = f_i(\mathbf{X}_{pa(i)}) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma_i^2) \tag{12}$$

  where the function f_i is sampled from a GP prior $f_i \sim \mathcal{GP}(0, k_i(\cdot, \cdot))$. The kernel k_i is a composition of a nonlinear radial basis function (RBF) and a linear kernel:

$$k_i(\mathbf{x}, \mathbf{x}') = \sigma_{f,i}^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell_i^2}\right) + \sigma_{L,i}^2 \langle \mathbf{x}, \mathbf{x}' \rangle \tag{13}$$

where ℓ_i denotes the lengthscale, while $\sigma^2_{f,i}$ and $\sigma^2_{L,i}$ represent the signal variances for the nonlinear and linear components, respectively. For root nodes ($pa(i) = \emptyset$), X_i is sampled from $\mathcal{N}(0, 1)$.

It should be noted that the results for GES on ER(3) graphs with 30 and 40 variables are not reported. As the algorithm either exceeded our predefined time threshold of 1000 seconds or encountered runtime failures during the search process. Such limitations underscore the inherent computational challenges faced by combinatorial optimization methods in high-dimensional causal discovery.

D.3 Sensitivity Analysis and Ablation Experiments

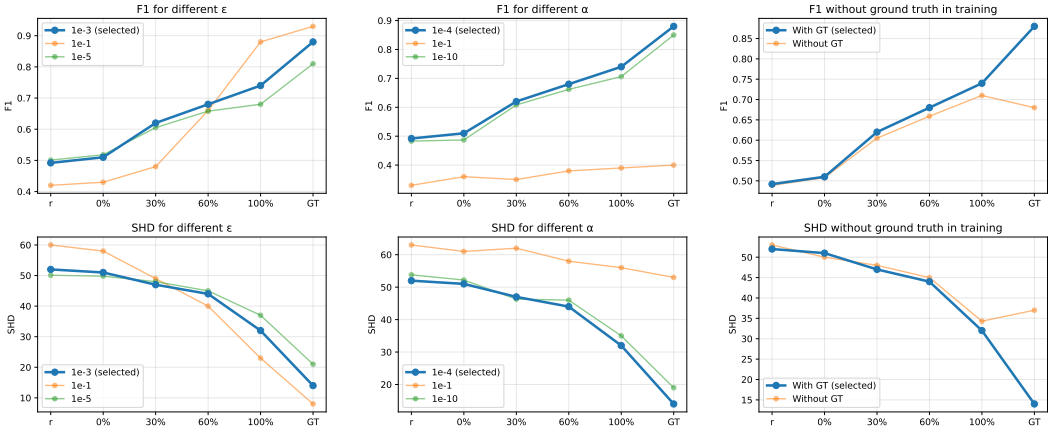


Figure 9: Sensitive Analysis for different parameters and ground truth anchor.

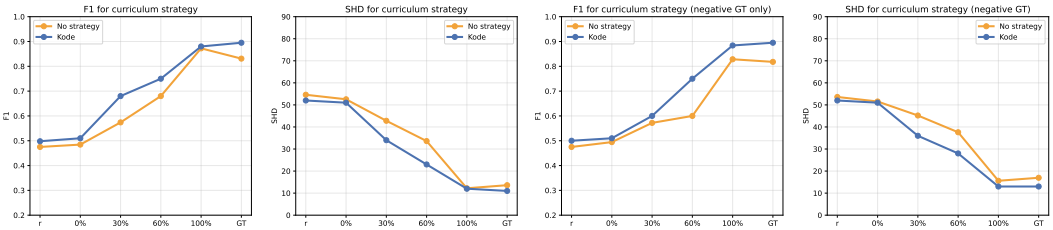


Figure 10: Ablation for curriculum strategy.

To clearly illustrate the model's sensitivity to different parameter and prior settings, we take as an example the Gaussian Processes mechanism with 20 variables and ER(3) density. Fig. 9 shows performance. Experiments covers interaction strength ϵ in decoder; strength α of the similarity regularizer in graph likelihood objective and ground truth proportion in training data.

We first visualize three curves for different values of ϵ (the first column subplots) . A large interaction strength (e.g., 10^{-1}) makes the model overly reliant on prior knowledge, encouraging a lazy shortcut behavior that weakens data-driven reasoning when knowledge is empty or sparse; while an excessively small value (e.g., 10^{-5}) limits effective knowledge utilization and weakens performance when knowledge is sufficient. The results suggest that an intermediate weak interaction strength (e.g., 10^{-3}) achieves the balance.

The second column subplots illustrate performance under different values of α. A large interaction strength (e.g., 10^{-1}) causes the model to overemphasize semantic alignment, leading to a collapse in performance. In contrast, when the strength is too low (e.g., 10^{-10}), the model cannot achieve optimal performance. As a compromise, we set $\alpha = 10^{-4}$, which preserves semantic similarity alignment without undermining the model's main task objective.

The subplots in the third column show the consequence without anchoring to ground truth. The performance drop is mainly observed under dense knowledge and ground-truth graph settings. Under

dense knowledge settings, performance declines noticeably. Even when the ground-truth graph is provided, the model does not appear to fully respect the structural constraints encoded by the true graph, preventing it from achieving optimal performance under complete ground-truth knowledge. The results suggest that, when given the ground-truth graph, the model primarily exploits the density of the provided knowledge, rather than interpreting each piece of knowledge as a constraint that should be strictly preserved or excluded. Consequently, its performance falls between that obtained with 60% and 100%reachability matrix, indicating that the model has not yet learned to faithfully enforce fully specified structural knowledge.

Fig. 10 illustrates how the model performance degrades when the curriculum learning strategy is removed. Under random-prior and no-prior settings(0%), the model essentially operates in a purely data-driven manner, yielding comparable F1 and SHD scores with and without curriculum learning. As the amount of prior knowledge increases, however, the model trained with curriculum learning consistently achieves substantially better performance. We hypothesize that the curriculum learning strategy enables the model to better exploit sparse prior knowledge while progressively developing a stronger understanding of dense structural knowledge.

These experiments use extreme cases to show the impact of overly high or low settings. We believe that careful finetuning could further improve performance and identify better parameters.

D.4 Performance with Prior Knowledge

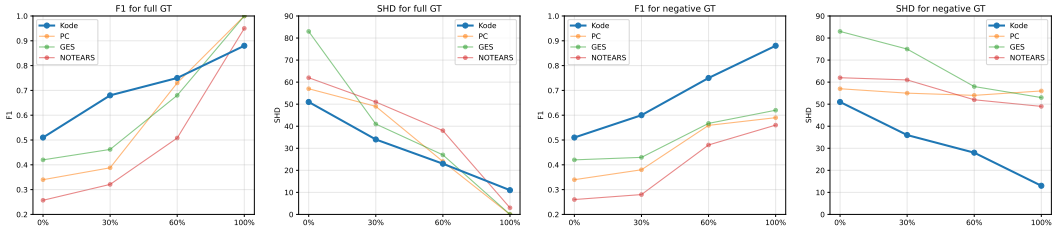


Figure 11: Performance with Prior Knowledge.

This section aims to demonstrate the performance of different methods to leverage prior knowledge, as illustrated in Fig. 11. Traditional causal discovery algorithms can exploit partial ground-truth graphs as prior knowledge by initializing from a given graph structure, thereby reducing the search space(Chickering, 2002; Spirtes & Glymour, 1991) and optimization space(Zheng et al., 2020), where required and forbidden edges can both be specified. However, as discussed in the main text, such strict constraints limits their practicality in real-world applications. Unlike our method, these baselines cannot model and benefit from knowledge such as *may-cause* relations as soft guidance and forcibly treating may-cause edges and random graph as hard inputs would inevitably introduce severe misleading signals. Therefore, for fairness, experiments are conducted under different missing rates of the ground-truth graph, with the interfaces provided by the gCastle library(Zhang et al., 2021). As shown in the figure, we compare performance under both the full ground-truth graph and the setting where only forbidden edges are provided. As in the previous subsection, we take as an example the Gaussian Process mechanism with 20 variables and ER(3) density.

The results show that the performance of all algorithms improves as more knowledge is provided. Our method maintains a clear advantage in the purely data-driven setting. However, as the amount of knowledge increases, the advantage of our method over PC and GES gradually narrows. Since the data-generation mechanism does not satisfy the identifibility assumptions of NOTEARS, our method still retains a substantial margin over it. When 100% knowledge is given as the full ground-truth graph, every edge is labeled as either required or forbidden, under which the baseline methods achieve a clear advantage. Nevertheless, apart from this extreme case of 100% ground truth, our method still outperforms the other baselines. On the other hand, when only negative knowledge, i.e., forbidden edges, is provided, our model can still benefit from this guidance and continues to improve. In contrast, the gains achieved by baseline algorithms become limited.

Experiments demonstrate our capability of exploiting precise knowledge and validating both the superiority of our knowledge encoding and the consistently strong performance of the model.

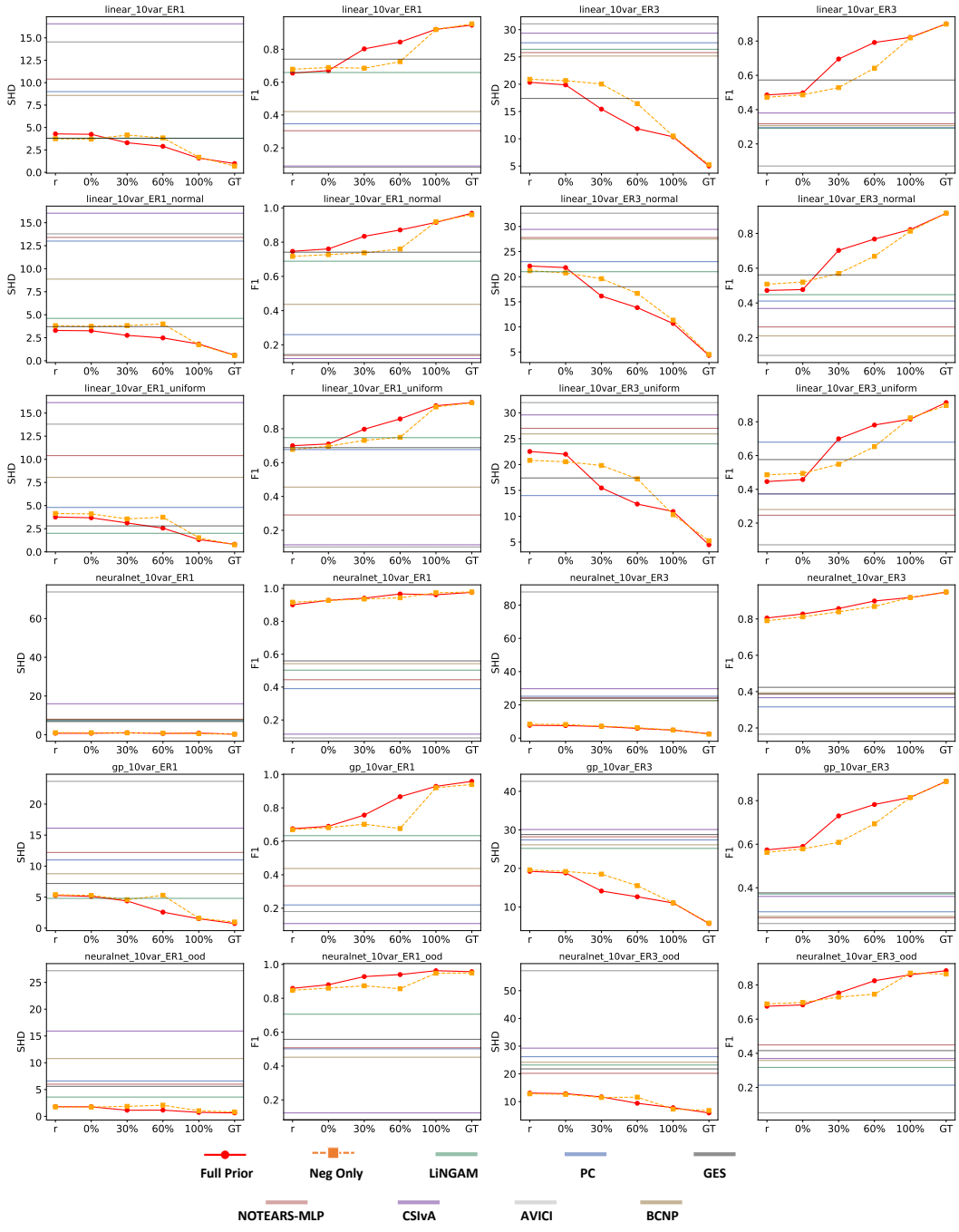


Figure 12: Results on 10 nodes.

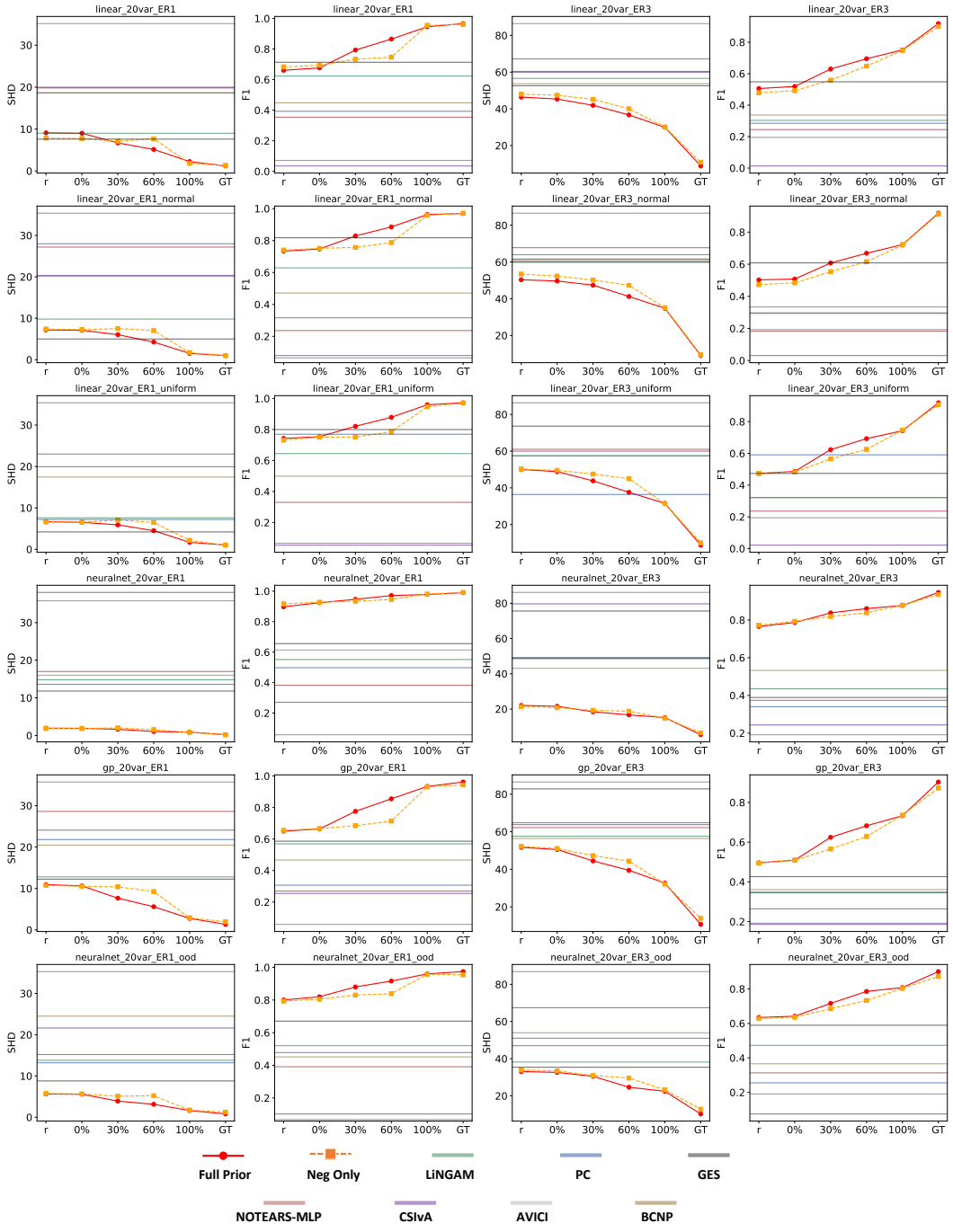


Figure 13: Results on 20 nodes.

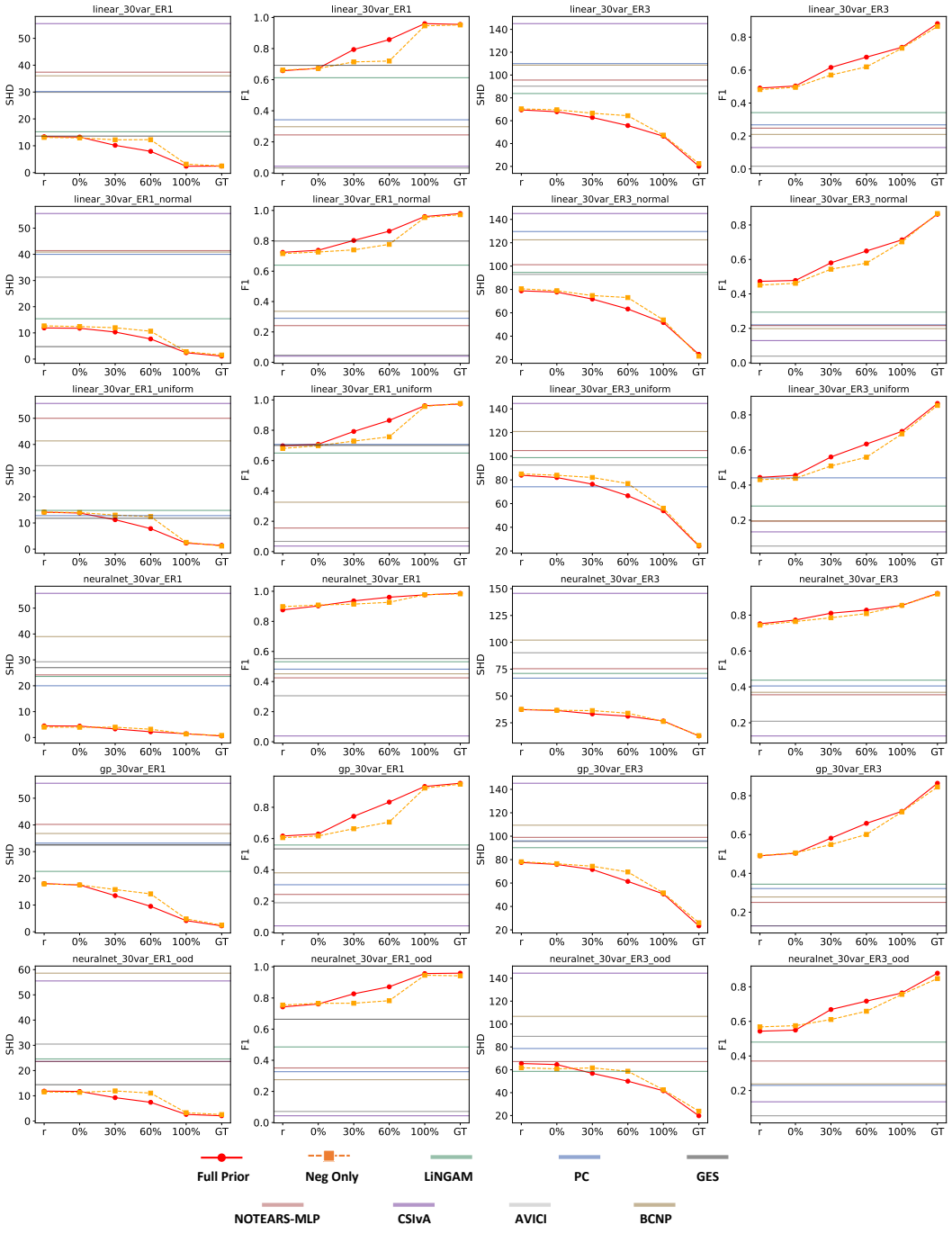


Figure 14: Results on 30 nodes.

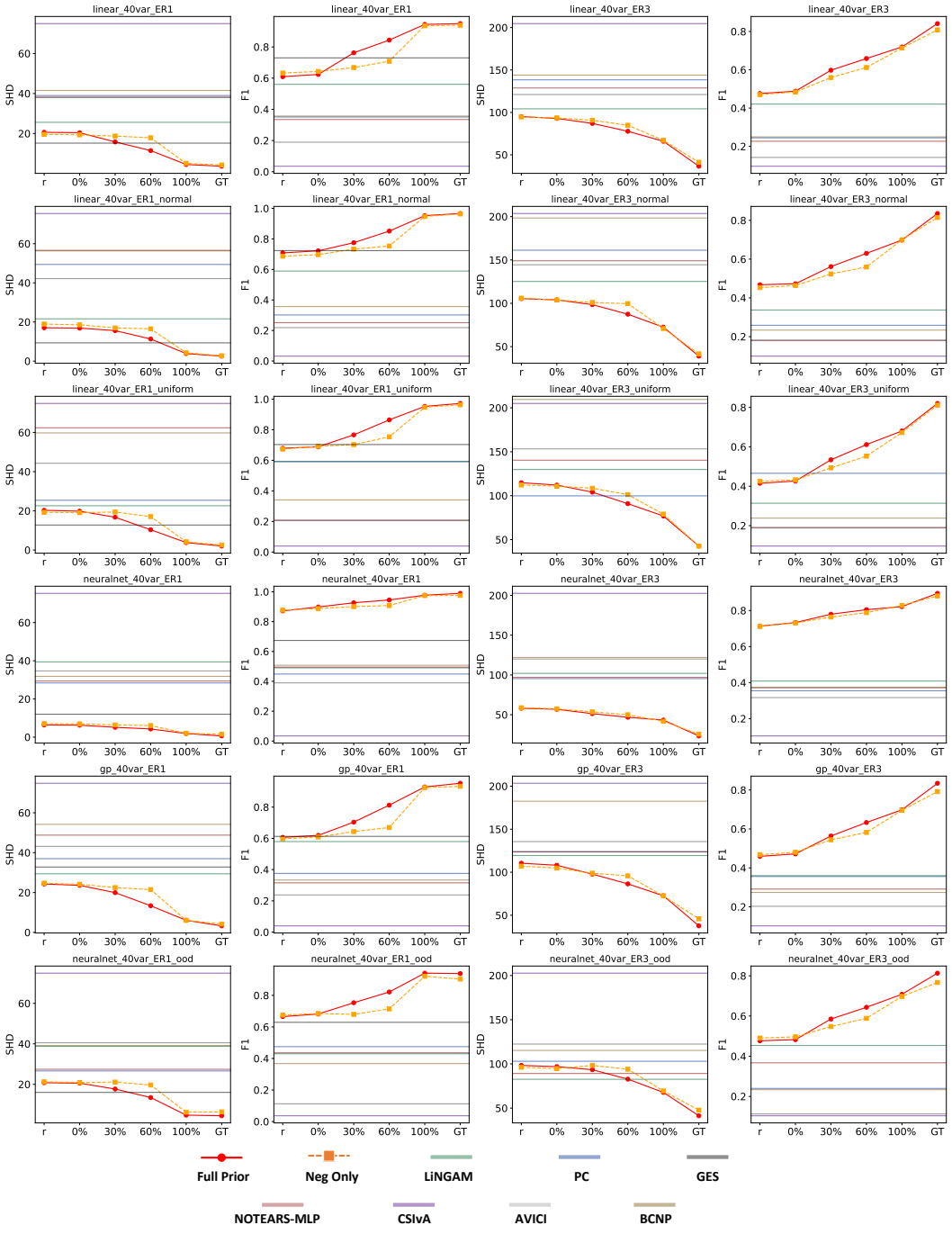


Figure 15: Results on 40 nodes.