Continuous Structure Constraint Integration for Robust Causal Discovery

Lyuzhou Chen[1] **Taiyu Ban**[1] **Derui Lyu**[1] **Yijia Sun**[1] **Kangtao Hu**[1] **Xiangyu Wang**[1†] **Huanhuan Chen**[1†]
[1]University of Science and Technology of China

Abstract

Causal discovery aims to infer a Directed Acyclic Graph (DAG) from observational data to represent causal relationships among variables. Traditional combinatorial methods search DAG spaces to identify optimal structures, while recent advances in continuous optimization improve this search process. However, integrating structural constraints informed by prior knowledge into these methods remains a substantial challenge. Existing methods typically integrate prior knowledge in a hard way, demanding precise information about causal relationships and struggling with erroneous priors. Such rigidity can lead to significant inaccuracies, especially when the priors are flawed. In response to these challenges, this work introduces the Edge Constraint Adaptive (ECA) method, a novel approach that softly represents the presence of edges, allowing for a differentiable representation of prior constraint loss. This soft integration can more flexibly adjust to both accurate and erroneous priors, enhancing both robustness and adaptability. Empirical evaluations demonstrate that our approach effectively leverages prior to improve causal structure accuracy while maintaining resilience against prior errors, thus offering significant advancements in the field of causal discovery.

1 INTRODUCTION

Causal discovery focuses on inferring directed acyclic graphs (DAGs) from observational data to represent relationships between variables (Pearl, 1995; Greenland et al., 1999; Kalisch and Bühlman, 2007), playing a pivotal role in numerous domains (Sanford and Moosa, 2012; Zhang et al., 2013). Causal discovery often demands large-scale and high-quality data. Fortunately, incorporating prior knowledge can mitigate these requirements, significantly enhancing the accuracy of causal structure identification (Amirkhani et al., 2016; Constantinou et al., 2023). The influence of prior knowledge extends beyond direct constraints on local structures to encompass global DAG structure (Li and Beek, 2018; Ban et al., 2023a), which is crucial in practice as integrating existing causal knowledge elucidates previously unknown causal mechanisms.

Traditionally, causal discovery methods explore the space of DAGs to identify the optimal structure (Chickering, 2002; Wang et al., 2021). While this search process is challenging, integrating structural constraints informed by prior knowledge is conveniently facilitated. Recently, a class of methods based on continuous optimization has emerged, which avoids complex search processes by leveraging machine learning techniques (Zheng et al., 2018; Yu et al., 2019; Zhu et al., 2019). However, few studies have explored how to effectively integrate these continuous optimization methods with prior knowledge.

Traditionally, causal discovery methods explore the space of DAGs to identify the optimal structure, which is challenging but can conveniently integrate structural constraints informed by prior knowledge (Chickering, 2002; Wang et al., 2021). Recently, continuous optimization methods have emerged, using loss functions to model acyclic constraints and relying on optimizers instead of complex search processes (Zheng et al., 2018; Yu et al., 2019; Zhu et al., 2019). These methods avoid the traditional search and pruning challenges posed by acyclic constraints and the large search space typical of combinatorial methods. However, effectively integrating these continuous optimization approaches with prior knowledge remains relatively unexplored.

Although some efforts have attempted to use priors that constrain edges in continuous optimization (Chen and Ge, 2023; Sun et al., 2023; Wang et al., 2024; Liang et al., 2023; Ramsey et al., 2018), they typically satisfy priors by specifying ranges for causal relationship weights, which often requires excessive information about the true weight range. In terms of performance, this approach, which can also be called hard constraints, does not smoothly characterize the

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258. †Corresponding author.

extent to which candidate DAGs satisfy the constraints, impairing the fine distinction between different weights, such as those at the margins of value ranges and those within. Consequently, hard constraints struggle to direct optimization during learning. Besides this, the performance of hard-constrained methods is also limited by the quality of the prior. When the prior contains errors, hard constraints cannot robustly or automatically judge the trade-offs of the prior, which may introduce serious errors and even destroy the learned correct causal structure.

To address these challenges, this work introduces a soft Edge Constraint Adaptation (ECA) method for continuous structural learning. Specifically, it employs causal weights to softly represent the presence of edges, smoothing the learning of edges and achieving a differentiable representation of prior constraint loss. Building on this, the proposed method robustly and adaptively utilizes priors through the competitive interplay between prior constraint loss and the original objective function, which act as proxies for the judgments of priors and data on the causal structure, respectively. When the causal structure demanded by priors overly contradicts the data, the objective function suppresses the corresponding structural modifications. Conversely, when the causal structure specified by priors is corroborated by the data, the prior constraint loss facilitates the appropriate structural adjustments.

Our contributions are:

- Proposal of a smooth ECA function, integrating the prior edge constraints for continuous DAG structure learning in a soft way, which is the first in this domain.
- The tolerance of the ECA loss to prior errors, attributed to the trade-off between consistency of data and prior.
- Easy use of the ECA loss, allowing its direct application in diverse continuous optimization methods of causal discovery to enable the integration of prior knowledge.

2 RELATED WORK

2.1 The Development of Causal Discovery

Causal discovery aims to understand causal relationships between variables to improve prediction and decision-making accuracy (Rios et al., 2021). It employs various methods to discover the true Directed Acyclic Graph (DAG) from observational data. Constraint-based methods, such as PC (Spirtes et al., 2001) and FCI (Colombo et al., 2012), use conditional independence tests to infer DAG structures, while score-based methods assess the fit of a candidate DAG using scoring functions. Hybrid methods like MMHC (Tsamardinos et al., 2006) combine these approaches. As the latest progress, Kuipers et al. (2022) combines constraint-based independence tests with MCMC search (a kind of score-based method). This approach simplifies MCMC procedures using table lookups and includes corrections for initial test errors, significantly enhancing network learning efficiency and accuracy. Some studies also explore data attributes like non-Gaussianity and nonlinearity to analyze DAGs (Shimizu et al., 2006; Hoyer et al., 2008; Gretton et al., 2009; Peters et al., 2011; Ghoshal and Honorio, 2018; Khemakhem et al., 2021).

Zheng et al. (2018) revolutionized causal discovery by introducing NOTEARS, which converts the combinatorial search of DAGs into a continuous optimization challenge, allowing for deep learning optimizers to manage acyclic constraints more efficiently. Building on this innovation, recent research has further developed continuous optimization techniques for causal discovery (Chen et al., 2023). For example, Wei et al. (2020) enhanced the algebraic characterization of DAG acyclicity, and Ng et al. (2019) created GOLEM, utilizing soft acyclicity constraints to boost optimization. Additionally, Bello et al. (2022) introduced DAGMA, notable for its ability to detect longer cycles, provide better gradients, and achieve faster performance.

2.2 Prior-based Causal Discovery

Integration of prior knowledge in causal discovery has been extensively studied in the realm of combinatorial DAG optimization. Common structural constraints investigated include edge existence (or absence) (De Campos and Castellano, 2007), ancestral constraints (Chen et al., 2016; Ban et al., 2023b), and ordering constraints (Ma et al., 2017). Our focus in this paper is primarily on the straightforward yet crucial constraints of edge existence and absence.

The methods for integrating prior knowledge are broadly classified into 'hard' and 'soft' approaches. The hard approach adheres strictly to every structural constraint, prioritizing constraint fulfillment over data consistency (Li and Beek, 2018). For example, Hyttinen et al. (2014, 2016) formulated the causal discovery method as a constraint satisfaction problem and solved it using multiple solvers including answer set programming. On the other hand, the soft approach finds a balance between data consistency and constraint adherence, often by extending the scoring criteria to include structural constraints with an assigned prior confidence (O'Donnell et al., 2006; Borboudakis and Tsamardinos, 2014). This approach is notably tolerant to errors in prior knowledge. Our study is inspired by and expands upon the principles of the soft approach.

However, the integration of prior knowledge in continuous optimization for causal discovery remains an area worth exploring. To the best of our knowledge, current approaches to related problems mainly implement priors by fixing upper and lower bounds on causal weights (Chen and Ge,

2023; Wang et al., 2024; Liang et al., 2023). For example, the method proposed by Sun et al. (2023) and Ramsey et al. (2018) constrains the bounds to integrate prior knowledge using user-provided lower and upper bounds. The work by Sun et al. (2023) focuses on learning dynamic Bayesian Networks, constraining edge existence or absence by freezing corresponding parameters during training. However, these methods, as hard methods, have limitations. They cannot accommodate prior errors and have difficulty in assisting the learning of causal weight values.

There are also some methods that cover the prior constraints by extending the objective function. For example, Hasan and Gani (2022) optimizes the number of priors that are not satisfied, Wang et al. (2024) optimizes the number of causal weights below the target threshold, and Bello et al. (2022) directly modifies the gradient of causal weights based on the prior during the optimization process. However, these methods only characterize whether the prior is satisfied.

In fact, for the continuous scenario which uses functions to model relationships, different edge weights correspond to different influences, which is crucial for optimizing the objective function and downstream tasks. Thus, even with the same edge prior, continuous scenarios require careful consideration of how priors are applied. In contrast, the method proposed in this paper softly represents the existence of edges, thereby smoothing the learning of edges. At the same time, the proposed method uses the competition between the prior constraint loss and the objective function to use the prior robustly and adaptively.

3 PRELIMINARIES

This section introduces the foundational concepts of causal discovery, covering the underlying task, graphical model, continuous optimization, and integration of prior.

3.1 Causal Discovery

Definition 1 (Causal Discovery from Observed Data)**.** *Given the observed data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ consisting of n independent and identically distributed samples from the joint distribution $P(X)$ of a d-dimensional random vector $X = (X_1, \ldots, X_d)$, the objective of causal discovery is to infer the relationships between these variables by constructing a Bayesian network, whose structure is a DAG $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ consisting of d nodes. Each edge in E corresponds to the direct dependency between a pair of variables, and the distribution of a variable is determined by its parent variables.*

According to the chain rule, the joint distribution can be decomposed into a product of conditional distributions in multiple trivial ways such as using a complete DAG, which is not our goal. We aim for the resulting DAG to possess a "minimal property", implying a need for a more restrained representation. Therefore, we assume that the joint distribution $P(X)$ is faithful to the DAG G, meaning there are no additional independencies in $P(X)$ other than those implied by G (Pearl, 2009). Furthermore, to ensure that the DAGs effectively capture causal relationships, we also assume that the distribution $P(X)$ is Markov with respect to G, which means that a variable is conditionally independent of its non-descendants given its parents in the DAG.

Definition 2. *The joint probability distribution of a random variables set $X = \{X_1, X_2, ..., X_d\}$ is said to be Markov with respect to DAG G if it can be decomposed into a product of conditional probability distributions, each corresponding to a variable conditioned on its parents in the G. Formally, the distribution $P(X)$ can be expressed as:*

$$P(X) = \prod_{i=1}^{d} P(X_i \mid Pa_G(X_i))$$

where $Pa_G(X_i)$ is the set of parent nodes of X_i in G.

The DAG learned from observed data is invaluable for uncovering the underlying dependencies and understanding potential causal mechanisms (Korb and Nicholson, 2010). Prevailing models in this field include Discrete Bayesian networks (Pearl, 2009) and Structural Equation Models (SEMs) (Spirtes et al., 2000), applicable to discrete and continuous data, respectively.

3.2 Structural Equation Model

We use Structural Equation Model (SEM) to model the specific relationships between random variables (Pearl, 2009):

Definition 3 (Structural Equation Model)**.** *Each variable X_i in random variable set X is modeled as a function of other random variables:*

$$X_i = f_i(X) + \epsilon_i, \quad \forall i \in [d] \tag{1}$$

where the notation $[d]$ to represent the set of integers $\{1, 2, \ldots, d\}$. ϵ_i represents noise with an expected value of zero, i.e., $E(\epsilon_i) = 0$. Furthermore, we assume ϵ_i is independent and uncorrelated with any variables. The function $f : \mathbb{R}^d \to \mathbb{R}$ might be nonlinear or nonparametric (i.e. the relationship between parent and child nodes may be too complex to express with a simple function class). Note that although each f_i uses X as its input, it actually depends on only certain components of X that are the causes of X_i. Specifically, f_i does not rely on X_i itself, in alignment with the physical premise that a variable does not influence itself. Therefore, the Equation 3 can be rewritten as:

$$X_i = f_i(Pa_G(X)) + \epsilon_i, \quad \forall i \in [d] \tag{2}$$

When the function set $f = (f_1, f_2, \ldots, f_d)$ consists only of linear functions, it is analogous to a weighted adjacency matrix W, which is the goal of causal discovery. At this time, the Equation 3 becomes:

$$X_i = X \cdot W_{\cdot i} + \epsilon_i, \quad i \in [d] \tag{3}$$

The "$\cdot$" between variables represents matrix-vector multiplication, and the subscript "$\cdot$" represents a row or column.

When f is nonlinear, a DAG structure can also be used to characterize the dependencies between variables. Under these conditions, $W_{i,j}$ is defined as the norm of the partial derivative $\partial_i f_j$ with respect to X_i, expressed as $W_{i,j} = \|\partial_i f_j\|_2$. It can be seen that W defined in this way is an extension of the linear f case, which serves as a weighted adjacency matrix, with zeros on its diagonal. In addition, $W_{i,j} = 0$ means f_j and X_i do not depend on each other, while a higher $W_{i,j}$ value suggests a stronger influence of X_i on f_j. Consequently, W effectively represents the dependency among the variables $(X_1, X_2, \ldots, X_d)$, which is the goal of causal discovery in nonlinear situations.

3.3 Continuous Optimization of Causal Discovery

To enforce acyclicity constraints, many traditional researchs treat causal discovery as a combinatorial problem. The work by Zheng et al. (2018) pioneers a shift to a continuous optimization framework within the SEM context.

Definition 4 (Continuous Optimization of DAGs)**.** *Given the observed data $\mathbf{X} \in \mathbb{R}^{n \times d}$ on d random variables, the objective is to optimize the following function:*

$$\min_{W \in \mathbb{R}^{d \times d}} F(W; \mathbf{X}) \quad \textit{subject to } h(W) = 0 \tag{4}$$

where $F(\cdot)$ is the loss of the DAG given data $\mathbf{X}$, presented as follows in the SEM context:

$$F(W; \mathbf{X}) = \frac{1}{2n} \|\mathbf{X} - \mathbf{X}W\|_F^2 + \lambda \|W\| \tag{5}$$

where $\lambda\|W\|$ is the regularization term to ensure the sparse nature of the DAG. $\|\cdot\|_F^2$ is the square of the Frobenius norm of a matrix. $h(\cdot)$ is a function to characterize the DAG-ness of the graph:

$$h(W) = \operatorname{tr}\left(e^{W \circ W}\right) - d \tag{6}$$

where $\operatorname{tr}(e^A)$ is the trace of the matrix exponential to A:

$$\operatorname{tr}(e^A) = \operatorname{tr}(I) + \operatorname{tr}(A) + \frac{1}{2!}\operatorname{tr}(A^2) + \cdots \tag{7}$$

Each non-zero element of W indicates the presence of an edge. Broadly speaking, the (i, j)th element of W^k indicates the number of k-length path from X_i to X_j. A cycle is a path from a variable to itself, therefore, a graph does not have k-length cycle if and only if the trace $\operatorname{tr}(A^k)$ is 0. Therefore, the acyclicity is equivalent to $h(W) = 0$.

4 EDGE CONSTRAINT ADAPTIVE LOSS

This section develops the Edge Constraint Adaptive (ECA) loss function under the framework of SEM with Gaussian distributed noise. Consider a set of d continuous random variables $X = (X_1, \ldots, X_d)$, the observed data $\mathbf{X}$ and the edge constraints $\mathbf{\Pi}$ (described in section 4.1), our objective is to optimize:

$$\min_{W \in \mathbb{R}^{d \times d}} F'(W; \mathbf{X}, \mathbf{\Pi}) \quad \text{subject to } h(W) = 0 \tag{8}$$

where function F' is a differentiable representation of the loss concerning both observed data and prior constraints, which will be described and derived from the likelihood function of DAGs in this section.

4.1 Definition of Prior Edge Constraints

The prior constraints on edges are defined as $\mathbf{\Pi} = (W_m, W_p)$, where $W_m \in \{0, 1\}^{d \times d}$ is a mask matrix and $W_p \in [0, 1]^{d \times d}$ represents a confidence matrix related to prior constraints. In the mask matrix W_m, an element $(W_m)_{ij}$ being 1 indicates a prior constraint is applied to the corresponding edge (X_i, X_j). In the confidence matrix W_p, each element $(W_p)_{ij}$ specifies the prior confidence of the edge existence (X_i, X_j). It is important to note that $(W_p)_{ij}$ comes into play if and only if $(W_m)_{ij} = 1$.

This formulation of prior edge constraints is comprehensive. For a constrained edge (X_i, X_j) (where $(W_m)_{ij} = 1$), the value of $(W_p)_{ij}$ suggests the probability of its existence in the graph. A value in the range $(0.5, 1]$ indicates a tendency towards the edge's presence, with higher probabilities reflecting greater confidence. Conversely, a value in the range $[0, 0.5)$ implies the edge's absence, with lower probabilities indicating stronger confidence. Thus, the framework of $\mathbf{\Pi} = (W_m, W_p)$ allows for the representation of existence and absence constraints on any edge with varying degrees of confidence. In this paper, we assume that the events in which the candidate graph satisfies prior constraints $\mathbf{\Pi}$ can be decomposed into multiple events that satisfy a single prior (X_i, X_j).

4.2 Structural Likelihood with Prior Constraints

We develop a likelihood-based, differentiable objective for scoring graphs that incorporate prior edge constraints. Initially, let's examine the probability of a candidate DAG G given the observed data $\mathbf{X}$ and prior constraints $\mathbf{\Pi}$:

$$\begin{aligned} P(G \mid \mathbf{X}, \mathbf{\Pi}) &= \frac{P(\mathbf{X} \mid G, \mathbf{\Pi}) P(G \mid \mathbf{\Pi})}{P(\mathbf{X} \mid \mathbf{\Pi})} \\ &= \frac{P(\mathbf{X} \mid G) P(G \mid \mathbf{\Pi})}{P(\mathbf{X} \mid \mathbf{\Pi})} \end{aligned} \tag{9}$$

The second equality holds as the likelihood of the observed data remains unaffected by the prior constraints $\mathbf{\Pi}$ once

the DAG G is established. Given that $P(\mathbf{X}|\mathbf{\Pi})$ remains the same across all DAGs, our emphasis is on the *data-structure likelihood term*, associated with $P(\mathbf{X}|G)$, and the *prior-adherence term*, related to $P(G|\mathbf{\Pi})$. The specific formulation of the data-structure likelihood term is as follows:

$$\begin{aligned} P(\mathbf{X} \mid G) &= P(\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_d \mid G) \\ &= \prod_{i=1}^{d} P(\mathbf{X}_i \mid \mathrm{Pa}_G(\mathbf{X}_i)) \\ &= \prod_{i=1}^{d} \prod_{j=1}^{n} P(\epsilon_i = (f_i(\mathrm{Pa}_G(\mathbf{X}_i)))_j - \mathbf{X}_{ji}) \end{aligned} \tag{10}$$

Where $\mathrm{Pa}_G(\mathbf{X}_i)$ represents the value of X_i's parents in observed data $\mathbf{X}$ and $(f_i(\mathrm{Pa}_G(\mathbf{X}_i)))_j$ represents the predicted value of X_i obtained using these parents in j^{th} sample. The second equality holds due to the Markov property of the DAG (see Definition 2). The third equality stems from the definition of SEM (see Definition 3). Following this, the log-likelihood can be formalized as:

$$\begin{aligned} &\log P(\mathbf{X} \mid G) = \sum_{i=1}^{d} \sum_{j=1}^{n} \log P(\epsilon_i = f_i(\mathrm{Pa}_G(\mathbf{X}_i)) - \mathbf{X}_{ji}) \\ &= \sum_{i=1}^{d} \sum_{j=1}^{n} \log \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{(f_i(\mathrm{Pa}_G(\mathbf{X}_i)) - \mathbf{X}_{ji})^2}{2\sigma^2}) \\ &= nc_0 d - \frac{1}{2\sigma^2} \|\mathbf{f}(\mathrm{Pa}_G(\mathbf{X})) - \mathbf{X}\|_F^2 \end{aligned} \tag{11}$$

The second equality holds due to the Gaussian assumption of the noise. σ is the standard deviation of the Gaussian distribution, and $c_0 = \log \frac{1}{\sigma\sqrt{2\pi}}$ is a constant.

Now we consider the prior-adherence term:

$$\begin{aligned} &P(G \mid \mathbf{\Pi}) = \prod_{i=1}^{d} \prod_{j=1}^{d} \left(P_{ij}^G \mathbb{I}_{W_{ij} \neq 0} + (1 - P_{ij}^G) \mathbb{I}_{W_{ij}=0}\right) \\ &= P(G - \mathbf{\Pi}) \times \\ &\prod_{(W_m)_{ij}=1} \left(\mathbb{I}_{W_{ij} \neq 0} (W_p)_{ij} + \mathbb{I}_{W_{ij}=0} (1 - (W_p)_{ij})\right) \end{aligned} \tag{12}$$

where P_{ij}^G denotes the Bayesian prior that the edge (X_i, X_j) is in the true graph, representing the initial probability assumption about the existence or non-existence of the edge. W is the weighted adjacency matrix corresponding to candidate DAG G, $\mathbb{I}_{\texttt{condition}}$ is the indicator function valuing 1 if `condition` holds, and 0 otherwise. The prior constraint Π partially modifies the Bayesian prior P_{ij}^G, while $P(G - \mathbf{\Pi})$ represents the unmodified part of the probability, which is identical for all graphs. The first equality holds due to the assumption that the existence of edges is jointly independent in the structural distribution.

4.3 Soft Characterization for Edge Constraints

This section introduces a soft way to characterize the adherence between the graph structure and the given prior constraints. We begin by considering the logarithmic form of the prior-adherence term $P(G \mid \mathbf{\Pi})$, dropping the constant term $P(G - \mathbf{\Pi})$.

$$\begin{aligned} &\log P(G \mid \mathbf{\Pi}) \\ &= \sum_{(W_m)_{ij}=1} \log \left(\mathbb{I}_{W_{ij} \neq 0} (W_p)_{ij} + \mathbb{I}_{W_{ij}=0} (1 - (W_p)_{ij})\right) \\ &= \|W_m \circ \log\left(W_{\neq 0} \circ W_p + W_{=0} \circ (1 - W_p)\right)\|_\Sigma \end{aligned} \tag{13}$$

where $\circ$ denotes the Hadamard product, and $\|A\|_\Sigma$ denotes the sum of elements of A. $W_{\neq 0}$ and $W_{=0}$ are matrices indicating the presence or absence of edges. In the matrix $W_{\neq 0}$, the (i,j)th element is set to 1 if $W_{ij} \neq 0$ and 0 if $W_{ij} = 0$. Similarly, the (i,j)th element of $W_{=0}$ is 0 if $W_{ij} \neq 0$ and 1 if $W_{ij} = 0$. For a matrix $A = [a_{ij}]$, $\log(A)$ is the matrix of element-wise logarithms, $[\log(a_{ij})]$.

Clearly, the Equation (13) is a step function, with a discontinuity occurring at the point where $W_{ij} = 0$, where the value jumps. To make Equation (13) learnable for W, we use the following functions to approximate $W_{\neq 0}$ and $W_{=0}$:

$$W_{\neq 0} := |2S(W) - 1|, \; W_{=0} := 1 - W_{\neq 0} \tag{14}$$

where $S(\cdot)$ is the element-wise Sigmoid function. For $A = [a_{ij}]$, $S(A) = \left[\frac{1}{1+e^{-a_{ij}}}\right]$. This function transforms Equation (13) into a continuous function with W as the variable, smooth except at the point where $W_{i,j} = 0$. Furthermore, as the absolute value of W moves away from zero, the value of $W_{\neq 0}$ gradually increases, allowing $W_{\neq 0}$ to effectively and softly capture the presence of an edge. Figure 1 presents a visual illustration.

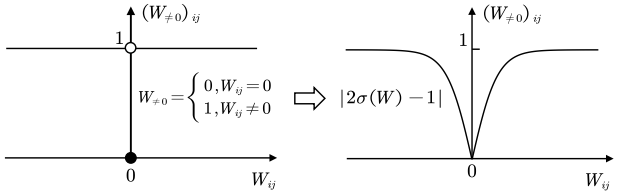

Figure 1: Soft characterization on the presence of edges.

Combining Equations (9), (11), (13) and (14), dropping the constant terms, we derive the scoring function of a DAG given observed data and prior constraints:

$$\begin{aligned} &F'(W; \mathbf{X}, \mathbf{\Pi}) \\ &= -\log P(\mathbf{X} \mid G) - \log P(G \mid \mathbf{\Pi}) \\ &= \frac{1}{\sigma^2} \left(\frac{1}{2} \|\mathbf{f}(\mathrm{Pa}_G(\mathbf{X})) - \mathbf{X}\|_F^2 - \sigma^2 b(W; \mathbf{\Pi})\right) \end{aligned} \tag{15}$$

where the expression of $b(W; \mathbf{\Pi})$ is as follows:

$$\begin{aligned} &b(W; \mathbf{\Pi}) \\ &= \|W_m \circ \log\left(W_{\neq 0} \circ W_p + (1 - W_{\neq 0}) \circ (1 - W_p)\right)\|_\Sigma \end{aligned} \tag{16}$$

and $W_{\neq 0}$ is defined by (14). For consistency with NOTEARS, we adopt the function within the parentheses of Equation (15) as the loss term. Furthermore, to facilitate the sparsity constraint, a regularization term $\lambda\|W\|$ is incorporated. This inclusion of the regularization term culminates in the final formulation of the ECA loss:

$$\begin{aligned} &F'(W;\mathbf{X},\mathbf{\Pi}) \\ =&\frac{1}{2}\|\mathbf{f}(\mathrm{Pa}_G(\mathbf{X}))-\mathbf{X}\|_F^2-\sigma^2 b(W;\mathbf{\Pi})+\lambda\|W\| \end{aligned} \tag{17}$$

Contrasting with the original loss presented in Equation (5), the ECA loss introduces an additional hyper-parameter σ and incorporates a prior-adherence term, $b(W;\mathbf{\Pi})$, which rewards the fulfillment of more edge constraints.

4.4 Optimization Details

Balance between data and prior. The first and second terms of Equation 17 represent the degree of fit of the candidate DAG structure to the data and the prior, respectively. However, in actual scenarios, the impact of data is often much greater than that of the prior. This is mainly because the value of the second term is limited, while the value of the first term is proportional to the sample size n. In order to maintain a balance between data and prior, we normalize the data to obtain a new F':

$$\begin{aligned} &F'(W;\mathbf{X},\mathbf{\Pi}) \\ =&\frac{1}{2n}\|\mathbf{f}(\mathrm{Pa}_G(\mathbf{X}))-\mathbf{X}\|_F^2-\sigma^2 b(W;\mathbf{\Pi})+\lambda\|W\| \end{aligned} \tag{18}$$

This can be considered as replacing the degree of fit of the structure to the sampled data with the degree of fit of the structure to the joint distribution.

Hyper-Parameter σ. The hyper-parameter σ represents the standard deviation of the noise term. In real-world scenarios, σ associated with each variable typically possesses a distinct value. However, for practicality, a uniform σ is commonly employed across all variables to simplify hyper-parameter tuning, bypassing the need for intricate SEM structural analysis (Ng et al., 2020).

In the ECA loss, σ affects the significance of prior edge constraints, thereby influencing the constraints' overall strength. For instance, with a smaller σ value (indicating a lower weight of the prior-adherence term), a larger prior probability is required to activate the edge constraint. In this case, the influence of the prior confidence on the constraint strength becomes prominent only at higher values. However, it's noteworthy that a prior confidence range of $[0.5, 1]$ (or $[0, 0.5]$) is generally able to span the entire spectrum of constraint strength, ranging from negligible consideration to full adherence (or inversely).

Optimization method. We employ the same optimization method and thresholding strategy as NOTEARS (Zheng et al., 2018) does. Concretely, the augmented Lagrangian method is used to enforce the acyclicity constraint, with the Proximal Quasi-Newton (PQN) method as an optimizer. As for deriving the graph G from the weighted adjacent matrix W, (X_i, X_j) is present in G when $|W_{ij}| > thr$, where thr is a pre-set positive number.

5 EMPIRICAL ESTIMATION

This section assesses the quality of DAGs derived via the proposed ECA method, which incorporates prior edge constraints. The assessment spans diverse scenarios characterized by variations in the number of prior constraints, types of constraints (existence or forbidden of edges), confidence levels in these constraints, and the prevalence of errors within the prior constraints.

Moreover, the study delves into the effects of prior refinement on previously undefined structures, particularly examining how the inclusion of prior information affects the precision in identifying structures that are not explicitly outlined in the priors. The ECA method's performance is also benchmarked against existing methods for incorporating priors to demonstrate its resilience against errors in prior.

Further, extensive testing is conducted in expanded contexts, including performance evaluations on larger, denser graphs, comparative analyses with conventional causal discovery methods, and assessments in environments affected by noise with non-equivalent Variance.

5.1 Experimental Setup

Datasets: The DAGs G, comprising d nodes and kd edges, are generated using the Erdös-Rényi (ER) model or scale-free (SF) model, denoted as ER-k or SF-k. The ER model creates edges between nodes with equal probability, while the SF model generates DAGs with power-law degree distributions. The number of nodes d tested ranged from 20 to 80. We conducted experiments with both linear and nonlinear models. For linear data generation, samples are formulated as $\mathbf{x} = \mathbf{x}W + \epsilon$ in $\mathbb{R}^d$, using Gaussian, Gumbel, or Exp noise and randomly assigned uniform edge weights W. For nonlinear data, the samples are generated using Mim method ($\mathbf{x} = \tanh(\mathbf{x}W_1) + \cos(\mathbf{x}W_2) + \sin(\mathbf{x}W_3) + \epsilon$) or MLP method, employing Gaussian noise with random uniform edge weights W_1, W_2, W_3. A total of n samples are produced by generating rows i.i.d. according to the selected model, resulting in observed data $\mathbf{X} \in \mathbb{R}^{n\times d}$. For the linear model, the sample sizes n range from $\{2d, 4d\}$, which will be referred to as medium size and large size below. For nonlinear model, n range from $\{20d, 40d\}$.

Prior Usage: The prior edge constraints include both edge existence and forbidden. A varying proportion of edges (not) present in the true DAG are randomly selected as edge (forbidden) existence constraints. To simulate imperfect

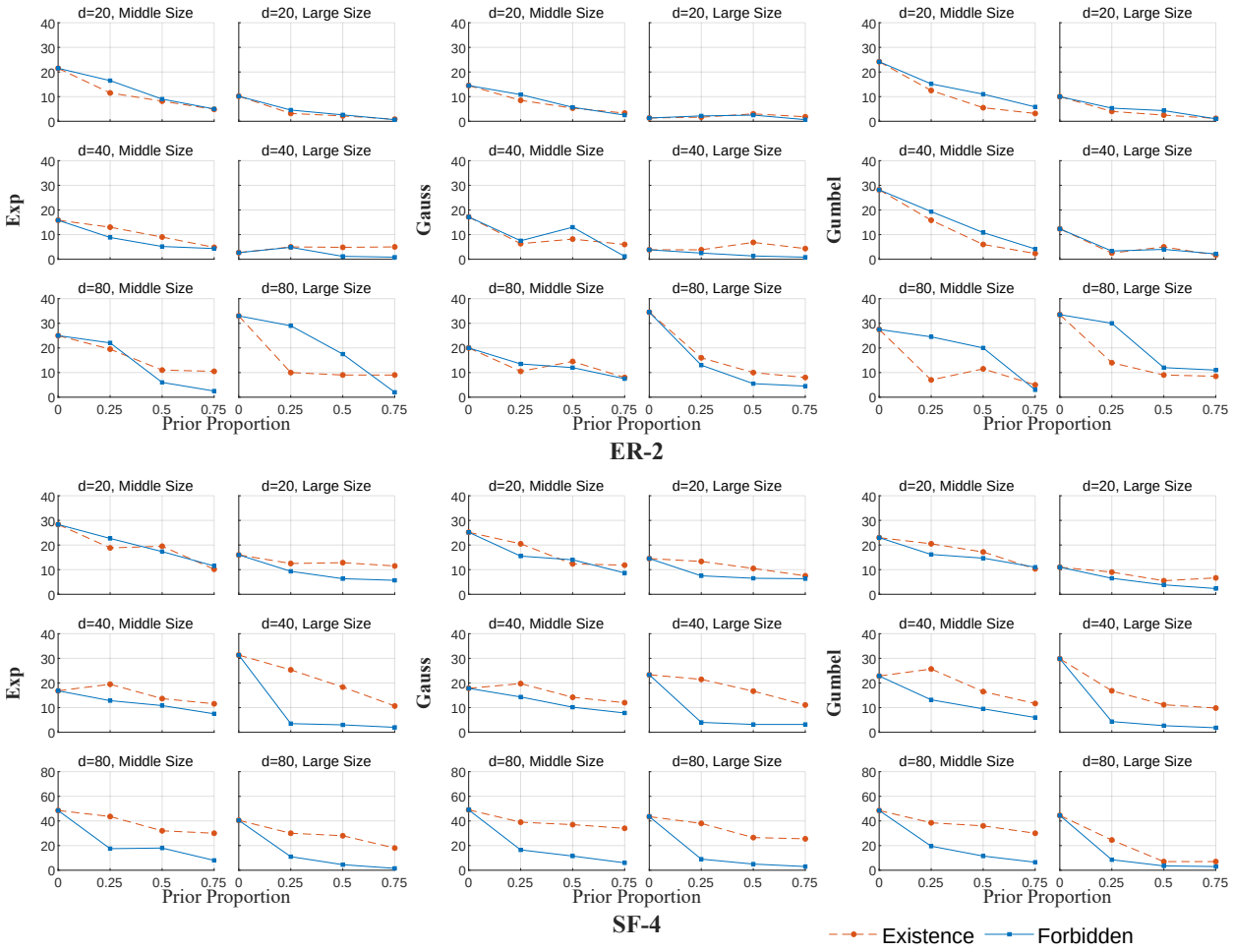

Figure 2: Performance of incorporating prior constraints in causal discovery on linear data. The y-axis represents the SHD of the method.

prior constraints on edge existence, extra edges or reversed edges that do not exist in the true DAG may be included as existing. Conversely, for simulating imperfect prior constraints on edge forbidden, edges that are actually present in the true DAG may be incorrectly treated as forbidden. The Ratio of Prior Errors to the true edge constraints is denoted as PER, with values in $\{0, 0.25, 0.5, 1.0\}$ (higher PER for lower prior quality).

For all experiments, the threshold for establishing edge presence, defined by $|W_{ij}| > thr$, is set to 0.3, and σ is fixed at 1.0. The confidence is assigned various values to observe different experimental scenarios. The normalization weight λ is adjusted in line with the prior confidence and quantity to maintain sparsity consistency with scenarios without priors, which is due to edge existence prior encourages more edges to be learned. The λ consists of a base value and a correction value. The base value is set according to the classic NOTEARS method, while the correction value is positively correlated with the number of edge existence priors and their credibility. Concretely, λ is set to 0.05 for edge forbidden. For edge existence, λ is set to $0.05 + \delta(pc, p)$ for linear data and $0.05 + \frac{1}{2}\delta(pc, p)$ for nonlinear data, where $\delta(pc, p) = p \times (pc - 0.5)$ with p standing for the proportion of constraints and pc for the prior confidence.

5.2 Improvement in Structure Learning

We evaluate DAGs learned using the NOTEARS alone and those learned with proposed ECA method under varying conditions of prior edge constraint types and quantities. For linear data, the results are shown in Figure 2. For nonlinear data, the results are shown in the Appendix.

The results indicate a consistent improvement in DAG quality when true prior constraints are incorporated. The degree of enhancement is directly proportional to the number of constraints applied. In some cases, the impact of priors on some datasets or configurations is limited, which may stem from an imbalanced ratio of missing to extra edges in the result graph relative to the true graph. The edge existence prior aims to guide the algorithm in identi-

fying previously unrecognized edges, whereas the edge forbidden prior highlights incorrectly identified edges. Since these functions are distinct and non-substitutable, the effectiveness of a certain class prior may reach its limits when the discrepancies between the result and true graph, i.e. mainly missing or extra edges, are small. Nevertheless, typically, either the edge existence or the edge forbidden prior exhibits improved performance. These observations hold true across different numbers of nodes and edges, sample sizes, and both linear and nonlinear data contexts.

5.3 Tolerance to Errors in Priors

This experiment evaluates the resilience of ECA loss against prior errors. Firstly, we examine the impact of varying error ratios and confidence on DAG quality, reporting the F1-score as the proportion of true edge existence constraints increases. The results are detailed in Figure 3 and there are following observations: (1) Despite a high error ratio, increasing the number of priors generally leads to improved DAG quality. (2) Increasing prior errors cause a decrease in DAG quality. (3) Higher confidence enhances performance for high-quality priors (PER=0) but is detrimental with low-quality priors (PER=1). (4) Nonlinear data is more sensitive to error ratios compared to linear data, under the same confidence settings.

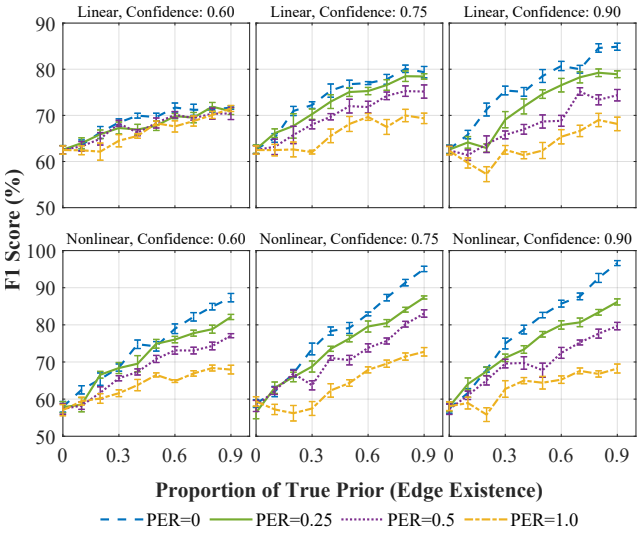

Figure 3: F1-score by ECA loss when prior errors are present. d=20, ER-2, n=40. PER is the ratio of prior errors to the true ones. pc is the prior confidence.

These observations align with theoretical expectations regarding the impact of prior errors. Notably, Observation 1 underscores ECA's tolerance capability, indicating its potential to enhance causal discovery even when prior errors match the correct ones in number. This is because the data itself reflects part of the true causal structure, and therefore, correct priors are generally more influential than incorrect ones. Observation 3 suggests that lower confidence settings offer a more robust defense against prior errors, albeit with reduced benefits from accurate priors. Observation 4 can be attributed to the increased difficulty in optimizing the data-structure likelihood term for nonlinear data, consequently elevating the influence of the prior-adherence term.

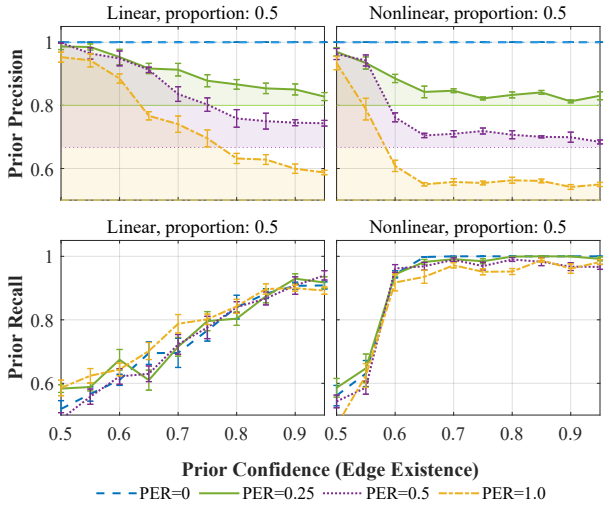

Figure 4: Precision and recall of accepted priors (d=20, ER-2, n=40).

Moreover, we present the precision and recall of accepted priors (edges both in the constraints and present in the learned DAG) with different confidences in Figure 4. The colored area in the precision results indicate the improved degree compared to the precision of the overall prior. We observe that the precision of the accepted prior by learning with ECA loss is always better than the precision of overall prior, specifically maintaining a high level when confidence is lower than 0.6. Moreover, the recall curve indicates that the nonlinear case is more prone to accept prior constraints than linear, aligning with the findings before.

5.4 Refinement of Unknown Structures

We evaluate the effectiveness of ECA in the refinement of unknown structures using accurate edge constraints. We detail the refined edges, i.e., discovered missing edges, removed extra edges, and corrected reversed edges, by classifying them into two parts, included (known) and not included (unknown) in the prior edge constraints. The results are reported in Figure 5. The blue line indicates the SHD by counting the refined edges included in the prior constraints, and the green area highlights the improvement on the quality of unknown structures (refined edges in total minus those in the prior constraints). The results underscore the vital role of prior knowledge in uncovering precise unknown causal relationships.

Table 1: Comparison of the effects of different prior constraint methods in causal discovery

PER	Node	20				40				60			
	Prior Proportion	0.3		0.6		0.3		0.6		0.3		0.6	
	Metric	F1	SHD	F1	SHD	F1	SHD	F1	SHD	F1	SHD	F1	SHD
0	NOTEARS_N*	83.4±6.0	14.5±5.5	83.4±6.0	14.5±5.5	89.3±5.3	17.2±8.5	89.3±5.3	17.2±8.5	82.4±7.7	40.0±18.4	82.4±7.7	40.0±18.4
0.2	ECA	**93.4±4.5**	**5.3±3.9**	**95.5±2.8**	**3.5±2.3**	**88.2±4.5**	**18.8±7.4**	92.5±1.0	12.7±1.8	**91.2±1.6**	**20.2±3.8**	**91.9±1.0**	**19.7±2.8**
	NOTEARS_G	83.6±3.3	13.7±2.5	85.7±3.4	11.3±2.7	85.8±2.6	22.7±3.8	89.8±2.6	16.2±4.1	87.4±1.4	28.7±2.7	85.8±1.1	32.5±2.4
	NOTEARS_B	73.7±3.4	24.5±4.1	81.0±2.6	17.0±2.4	92.3±3.8	12.2±6.0	92.7±1.5	12.3±2.5	92.3±0.9	17.5±1.9	90.8±0.6	22.7±1.4
0.4	ECA	**87.2±3.5**	**10.5±3.1**	**88.0±2.4**	**10.5±2.1**	85.8±4.1	23.5±6.8	**89.5±1.8**	**18.3±3.1**	**89.2±2.0**	**25.8±4.5**	**87.4±1.4**	**32.2±3.5**
	NOTEARS_G	79.8±4.5	17.0±3.9	85.2±1.6	11.8±1.5	84.9±2.5	24.0±3.6	88.3±3.8	18.5±6.0	86.8±1.0	30.0±1.9	85.0±1.4	34.5±3.1
	NOTEARS_B	68.1±3.2	30.2±4.5	79.2±4.3	17.8±3.9	87.5±4.1	20.5±6.7	80.6±0.7	34.7±1.5	89.0±1.2	26.3±2.7	85.2±0.4	39.3±1.0
0.6	ECA	**83.4±3.6**	**15.0±3.5**	**82.8±2.3**	14.7±2.3	**83.4±2.9**	**28.0±4.9**	**86.3±2.5**	**24.2±4.1**	**86.7±1.2**	33.7±3.0	83.2±1.5	43.8±4.3
	NOTEARS_G	78.8±2.9	18.3±2.6	81.8±3.6	14.2±2.6	83.5±3.2	26.5±5.0	87.9±3.6	19.2±5.6	86.2±1.1	31.3±2.3	83.9±1.1	37.2±2.5
	NOTEARS_B	70.7±2.6	25.0±2.9	69.1±3.2	26.2±3.1	83.5±4.0	28.3±7.1	62.1±0.8	79.3±3.3	86.6±1.7	34.0±4.0	75.8±0.9	66.5±2.3

* "NOTEARS_N" in this table represents the NETEARS method without prior.

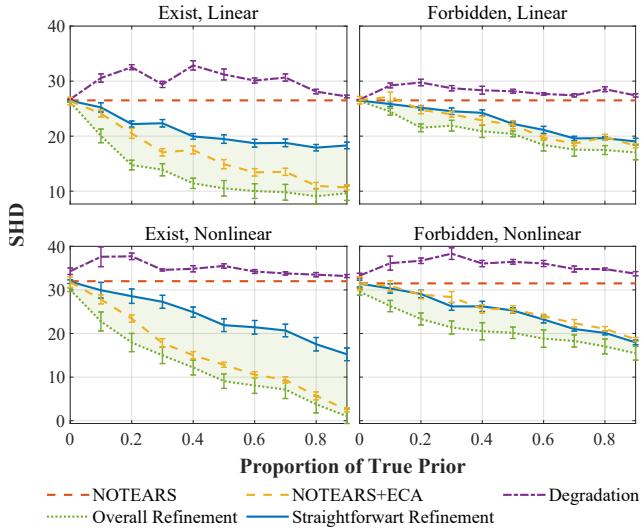

Figure 5: Separate display of improvements and damage on SHD under prior existence and forbidden constraints (d=20, ER-2, n=40). pc is the prior confidence.

5.5 Comparative Analysis of the Proposed Method and Other Methods

In this experiment, the proposed method is compared with currently proposed methods that combine priors. To ensure fairness, all methods use the continuous optimization method NOTEARS as the basis for causal discovery. Specifically, one method constrains the range of causal weights (Chen and Ge, 2023; Sun et al., 2023; Wang et al., 2024), called NOTEARS-B. Another method modifies the gradient, called NOTEARS-G (Bello et al., 2022). Our results that are statistically significantly better than other methods are highlighted. The results are shown in Table 1. Note that the "NOTEARS_N" line shows the basic results without priors.

It can be seen that the proposed method outperforms NOTEARS-B and NOTEARS-G in most cases. In particular, when the prior has a certain error, the proposed method can still improve the performance of causal discovery to a certain extent. At the same time, the PER boundary value of the available prior (the maximum PER when the performance is lower than the basic NOTEARS) of the proposed method is also greater than NOTEARS-B and NOTEARS-G, which highlights the quality of the prior of the proposed method.

6 CONCLUSIONS

In this paper, we introduce the Edge Constraint Adaptive (ECA) approach, a novel method for prior knowledge-based continuous optimization of causal discovery. This approach is directly extendable to related methods to integrate prior edge constraints. Our experiments demonstrate the effectiveness of this method in leveraging prior knowledge to improve DAG quality, its notable tolerance to prior errors, and its ability to refine the unknown structure. In summary, the ECA method not only equips existing continuous optimization strategies for causal discovery with a robust mechanism for applying prior knowledge but also lays the groundwork for future research in incorporating more complex forms of prior information.

Acknowledgments

This research is supported in part by the National Key R&D Program of China (No. 2021ZD0111700), in part by the National Nature Science Foundation of China (No. 62137002, 62176245, 62406302), in part by the Natural Science Foundation of Anhui province (No. 2408085QF195), in part by the Key Research and Development Program of Anhui Province (No. 202104a05020011), in part by the Key Science and Technology Special Project of Anhui Province (No. 202103a07020002), in part by the Fundamental Research Funds for the Central Universities under Grant WK2150110035.

References

Amirkhani, H., Rahmati, M., Lucas, P. J., and Hommersom, A. (2016). Exploiting experts' knowledge for structure learning of Bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2154–2170.

Ban, T., Chen, L., Lyu, D., Wang, X., and Chen, H. (2023a). Causal structure learning supervised by Large Language Model. *arXiv preprint arXiv:2311.11689*.

Ban, T., Chen, L., Wang, X., and Chen, H. (2023b). From query tools to causal architects: Harnessing large language models for advanced causal discovery from data. *arXiv preprint arXiv:2306.16902*.

Bello, K., Aragam, B., and Ravikumar, P. (2022). DAGMA: Learning DAGs via M-matrices and a Log-determinant acyclicity characterization. *Advances in Neural Information Processing Systems*, 35:8226–8239.

Borboudakis, G. and Tsamardinos, I. (2014). Scoring and searching over Bayesian networks with causal and associative priors. *arXiv preprint arXiv:1408.2057*.

Chen, E. Y.-J., Shen, Y., Choi, A., and Darwiche, A. (2016). Learning Bayesian networks with ancestral constraints. *Advances in Neural Information Processing Systems*, 29.

Chen, W., Qiao, J., Cai, R., and Hao, Z. (2023). On the role of entropy-based loss for learning causal structure with continuous optimization. *IEEE Transactions on Neural Networks and Learning Systems*.

Chen, Z. and Ge, Z. (2023). Directed acyclic graphs with TEARS. *IEEE Transactions on Artificial Intelligence*, 4(4):972–983.

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

Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321.

Constantinou, A. C., Guo, Z., and Kitson, N. K. (2023). The impact of prior knowledge on causal structure learning. *Knowledge and Information Systems*, pages 1–50.

De Campos, L. M. and Castellano, J. G. (2007). Bayesian network learning algorithms using structural restrictions. *International Journal of Approximate Reasoning*, 45(2):233–254.

Ghoshal, A. and Honorio, J. (2018). Learning linear structural equation models in polynomial time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pages 1466–1475. Proceedings of Machine Learning Research.

Greenland, S., Pearl, J., and Robins, J. M. (1999). Causal diagrams for epidemiologic research. *Epidemiology*, 10(1):37–48.

Gretton, A., Spirtes, P., and Tillman, R. (2009). Nonlinear directed acyclic structure learning with weakly additive noise models. *Advances in Neural Information Processing Systems*, 22.

Hasan, U. and Gani, M. O. (2022). KCRL: A prior knowledge based causal discovery framework with reinforcement learning. In *Machine Learning for Healthcare Conference*, pages 691–714. Proceedings of Machine Learning Research.

Hoyer, P. O., Shimizu, S., Kerminen, A. J., and Palviainen, M. (2008). Estimation of causal effects using linear non-Gaussian causal models with hidden variables. *International Journal of Approximate Reasoning*, 49(2):362–378.

Hyttinen, A., Eberhardt, F., and Järvisalo, M. (2014). Constraint-based causal discovery: Conflict resolution with answer set programming. In *UAI*, pages 340–349.

Hyttinen, A., Plis, S., Järvisalo, M., Eberhardt, F., and Danks, D. (2016). Causal discovery from subsampled time series data by constraint optimization. In *Conference on Probabilistic Graphical Models*, pages 216–227. PMLR.

Kalisch, M. and Bühlman, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8(3).

Khemakhem, I., Monti, R., Leech, R., and Hyvarinen, A. (2021). Causal autoregressive flows. In *Proceedings of International Conference on Artificial Intelligence and Statistics*, pages 3520–3528. PMLR.

Korb, K. B. and Nicholson, A. E. (2010). *Bayesian artificial intelligence*. CRC press.

Kuipers, J., Suter, P., and Moffa, G. (2022). Efficient sampling and structure learning of Bayesian networks. *Journal of Computational and Graphical Statistics*, 31(3):639–650.

Li, A. and Beek, P. (2018). Bayesian network structure learning with side constraints. In *International Conference on Probabilistic Graphical Models*, pages 225–236. Proceedings of Machine Learning Research.

Liang, J., Wang, J., Yu, G., Guo, W., Domeniconi, C., and Guo, M. (2023). Directed acyclic graph learning on attributed heterogeneous network. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10845–10856.

Ma, T.-Y., Chow, J. Y., and Xu, J. (2017). Causal structure learning for travel mode choice using structural restrictions and model averaging algorithm. *Transportmetrica A: Transport Science*, 13(4):299–325.

Ng, I., Ghassami, A., and Zhang, K. (2020). On the role of sparsity and DAG constraints for learning linear DAGs. *Advances in Neural Information Processing Systems*, 33:17943–17954.

Ng, I., Zhu, S., Chen, Z., and Fang, Z. (2019). A graph autoencoder approach to causal structure learning. *arXiv preprint arXiv:1911.07420*.

O'Donnell, R. T., Nicholson, A. E., Han, B., Korb, K. B., Alam, M. J., and Hope, L. R. (2006). Causal discovery with prior information. In *AI 2006: Advances in Artificial Intelligence: 19th Australian Joint Conference on Artificial Intelligence, Hobart, Australia, December 4-8, 2006. Proceedings 19*, pages 1162–1167. Springer.

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

Pearl, J. (2009). *Causality*. Cambridge university press.

Peters, J., Janzing, D., and Scholkopf, B. (2011). Causal inference on discrete data using additive noise models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2436–2450.

Ramsey, J. D., Zhang, K., Glymour, M., Romero, R. S., Huang, B., Ebert-Uphoff, I., Samarasinghe, S., Barnes, E. A., and Glymour, C. (2018). TETRAD—A toolbox for causal discovery. In *8th international workshop on climate informatics*, pages 1–4.

Rios, F. L., Moffa, G., and Kuipers, J. (2021). Benchpress: A scalable and versatile workflow for benchmarking structure learning algorithms. *arXiv preprint arXiv:2107.03863*.

Sanford, A. D. and Moosa, I. A. (2012). A Bayesian network structure for operational risk modelling in structured finance operations. *Journal of the Operational Research Society*, 63(4):431–444.

Shimizu, S., Hoyer, P. O., Hyvärinen, A., Kerminen, A., and Jordan, M. (2006). A linear non-Gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10).

Spirtes, P., Glymour, C., and Scheines, R. (2001). *Causation, prediction, and search*. MIT press.

Spirtes, P., Glymour, C. N., and Scheines, R. (2000). *Causation, prediction, and search*. MIT press.

Sun, X., Schulte, O., Liu, G., and Poupart, P. (2023). NTS-NOTEARS: Learning nonparametric DBNs with prior knowledge. In *International Conference on Artificial Intelligence and Statistics*, pages 1942–1964. Proceedings of Machine Learning Research.

Tsamardinos, I., Brown, L. E., and Aliferis, C. F. (2006). The Max-Min Hill-Climbing Bayesian network structure learning algorithm. *Machine Learning*, 65:31–78.

Wang, Z., Gao, X., Liu, X., Ru, X., and Zhang, Q. (2024). Incorporating structural constraints into continuous optimization for causal discovery. *Neurocomputing*, 595:127902.

Wang, Z., Gao, X., Yang, Y., Tan, X., and Chen, D. (2021). Learning Bayesian networks based on order graph with ancestral constraints. *Knowledge-Based Systems*, 211:106515.

Wei, D., Gao, T., and Yu, Y. (2020). DAGs with No Fears: A closer look at continuous optimization for learning Bayesian networks. *Advances in Neural Information Processing Systems*, 33:3895–3906.

Yu, Y., Chen, J., Gao, T., and Yu, M. (2019). DAG-GNN: DAG structure learning with graph neural networks. In *International Conference on Machine Learning*, pages 7154–7163. Proceedings of Machine Learning Research.

Zhang, B., Gaiteri, C., Bodea, L.-G., Wang, Z., McElwee, J., Podtelezhnikov, A. A., Zhang, C., Xie, T., Tran, L., Dobrin, R., et al. (2013). Integrated systems approach identifies genetic nodes and networks in late-onset Alzheimer's disease. *Cell*, 153(3):707–720.

Zheng, X., Aragam, B., Ravikumar, P. K., and Xing, E. P. (2018). DAGs with NOTEARS: Continuous optimization for structure learning. *Advances in Neural Information Processing Systems*, 31.

Zhu, S., Ng, I., and Chen, Z. (2019). Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   The mathematical setting and assumptions is described in detail in Section 3 and the algorithm is described in Section 4.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]
   The anonymized source code is included in the supplemental material.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   The full set of assumptions of all theoretical results is included in Section 3 and 4.
   (b) Complete proofs of all theoretical results. [Yes]
   Complete proofs of all theoretical results is included in 4.
   (c) Clear explanations of any assumptions. [Yes]
   Clear explanations of assumptions is included in 3.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   The code, data, and instructions is included in supplemental material.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   All the training details have been presented.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   The specific measure and error bars is clearly defined.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
   The computing infrastructure used is described.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   All assets are cited.
   (b) The license information of the assets, if applicable. [Yes]
   The license information of the assets is included in supplemental material.
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
   New assets is included in supplemental material.
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A ADDITIONAL EXPERIMENTS

A.1 DAG Learning Performance on Nonlinear Data

In this experiment, We evaluated the DAGs learned using only NOTEARS and those learned with the proposed ECA method under varying prior edge constraint conditions. The results for nonlinear data are shown in Figure 6.

In Figure 6, ER2 and SF4 refer to the types of models described in Section 4.1, where MLP represents Multi-Layer Perceptron, and MIM denotes the Mixed Instance Model in multi-task learning. Specifically, three random matrices W_1, W_2, and W_3 are generated in a similar manner, and data is generated using $X = \tanh(XW_1) + \sin(XW_2) + \cos(XW_3) + z$. "Exist" and "Forbidden" represent the methods of providing priors, where "Exist" indicates the presence of edge priors and "Forbidden" indicates the absence of edge priors. In each table, d represents the number of nodes; "Middle Size" and "Large Size" correspond to cases where the size is 2 and 4, respectively. The horizontal axis of the table represents the proportion of constraints, while the vertical axis represents the SHD (Structural Hamming Distance) between the learned results and the true results, with smaller SHD indicating better learning performance.

According to Figure 6, it is evident that in most cases, as the prior rate increases, both the "Exist" and "Forbidden" methods show some improvement in learning performance. However, there are instances where learning performance may decrease despite the prior rate increase, compared with linear situation (see Figure 7). This is because, in nonlinear cases, the result tends to be more sparse and provides a small improvement margin on edge forbidden priors, which diminishes the effectiveness of the forbidden priors. Nevertheless, based on the results with edge priors, the proposed method remains effective. Additionally, in specific cases (e.g., the second column of the first row and the second column of the fourth row), the performance improvement of the resulting DAG is minimal, as sufficiently good results have already been achieved under the baseline conditions.

A.2 Evaluation of ECA Method on Denser Graphs

We conduct experiments on denser graphs to demonstrate that the effectiveness of the proposed method is not limited to sparse graphs. Specifically, we use the standard $G(n, p)$ model to generate random graphs with densities of 0.1 and 0.2, corresponding to expected node degrees of 8 and 16, respectively, for graphs with 80 nodes. In addition to experiments conducted on ER and SF graphs, we vary data ratios and noise types to assess the robustness of our method. The results are shown in Table 2.

For evaluation, we use several performance metrics, including F1 score and SHD. Additionally, we introduce the Structure Intervention Distance (SID) metric to quantify the difference in the intervention distribution between the learned graph and the true graph. This metric provides further insight into the effectiveness of the learned graph in representing the true underlying causal structure, particularly in terms of intervention consistency. Similar to SHD, a lower SID score reflects superior performance.

The results show that our method consistently outperforms other approaches in terms of both F1 and SHD across varying graph densities and noise types. Notably, the introduction of priors significantly enhances the quality of causal structure learning, and the performance improvements are not contingent on the sparsity of the graph. This suggests that the proposed method is effective across a broad range of graph densities, reinforcing its versatility and robustness for practical applications. The results also indicate that our method's performance holds up even in denser settings, where traditional methods may struggle due to higher connectivity and more complex causal dependencies.

A.3 Comparison between Traditional Method

To evaluate the performance of the proposed method, we extended the experimental analysis by comparing it with several classical combinatorial approaches, highlighting its general advantages over existing techniques. The comparison includes a range of methods:

- PC (Peter-Clark): A constraint-based method that uses conditional independence tests to learn the structure of a Bayesian network.
- HC (Hill-Climbing): A score-based method that iteratively improves a candidate network by selecting the best scoring graph according to a chosen score function.
- MMHC (Max-Min Hill Climbing): A hybrid method combining conditional independence tests and score-based search to optimize network structure.
- Astar: An exact search method that uses a heuristic search strategy to find the optimal network structure.
- Minobsx: An approximate hybrid method based on local search that incorporates both observed data and expert knowledge, capable of efficiently handling various prior knowledge constraints to find near-optimal network structures.

Additionally, we compare our method with the variance-based approach VARSORT proposed by Reisach et al., which operates in the context of continuous optimization.

For each random graph used in the experiments, a 50% prior constraint is introduced. These priors are applied as hard constraints, primarily to guide the correct structure

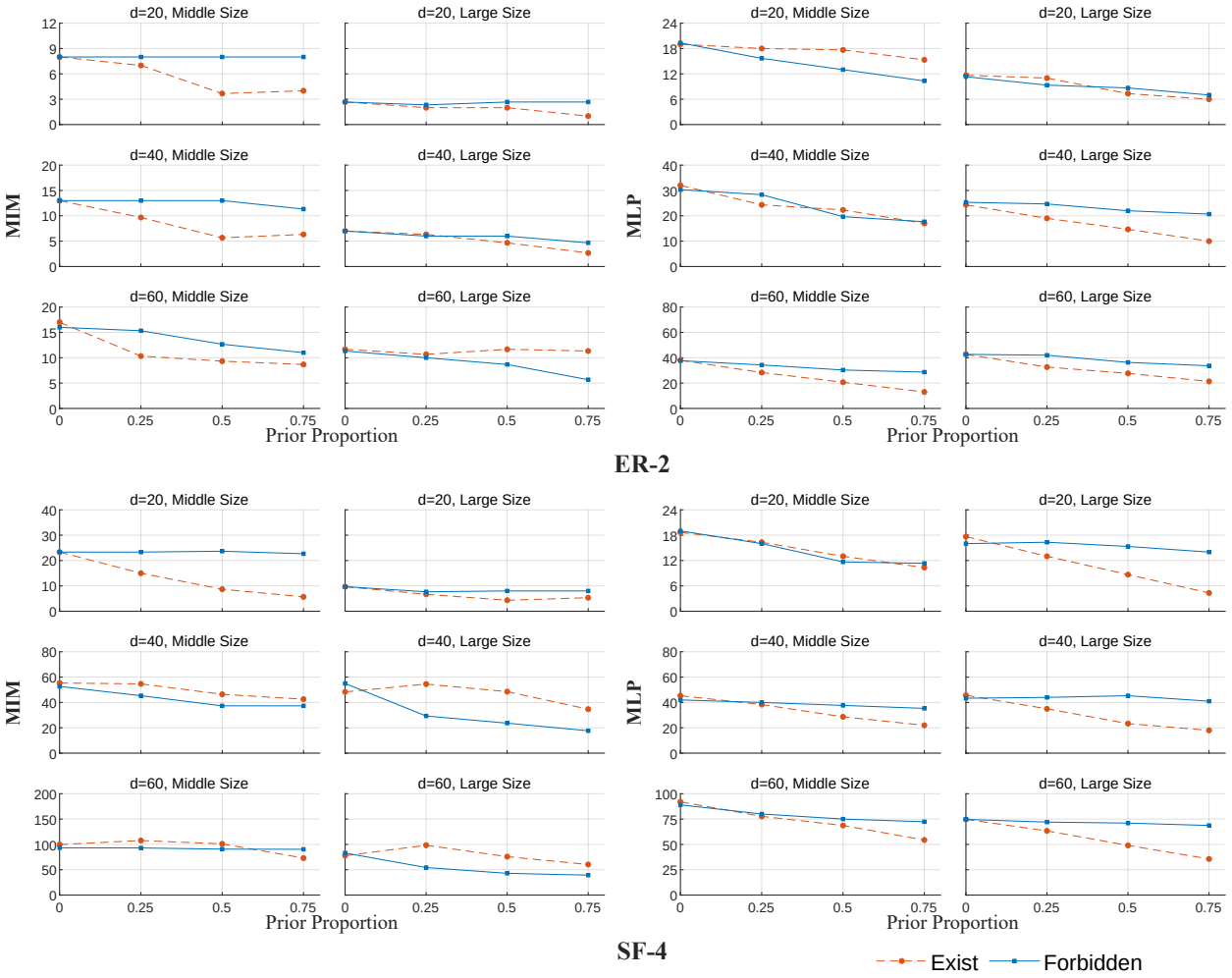

Figure 6: Performance of incorporating prior constraints in causal discovery on nonlinear data. The y-axis represents the SHD of the method.

and prune the search space. The effect of these priors varies depending on the method, but we focus on their application as hard constraints, as they significantly improve performance when the priors are correct. Although some of the comparison methods offer soft constraint implementations, we found that hard constraints yield more pronounced improvements in these cases.

The experiments were conducted on ER and SF graphs, with varying data sizes and noise types to assess the robustness of each method across different settings. The evaluation metrics include F1 score, SHD, and SID which are standard measures for structure recovery accuracy and structural discrepancy. The results are shown in Table 3.

The experimental results clearly demonstrate that our proposed method outperforms all other methods in terms of both F1 score and SHD, highlighting its effectiveness in accurately recovering the underlying structure of the graph. These two metrics, which focus on structure recovery and discrepancy, confirm the superior performance of our method in generating a graph that closely matches the true structure.

In terms of SID, our method ranks second overall, with the variance-based VARSORT method showing statistically superior performance. This can be attributed to VARSORT's better capacity to assess the node ordering under equal variance conditions, which enhances its ability to align the result graph more closely with the true intervention distribution. On the other hand, our method primarily reduces SID by minimizing the structural discrepancies between the learned and true graphs. While the SID metric does not always favor our method over VARSORT, it is important to note that, in many cases, our method significantly reduces the structural differences between the graphs. In fact, for certain datasets or configurations, the SID of our method is on par with or even outperforms the VARSORT method. This suggests that our method's ability to reduce structural discrepancies may sometimes compensate for its relatively lower performance in terms of intervention distribution alignment, making it a highly competitive approach

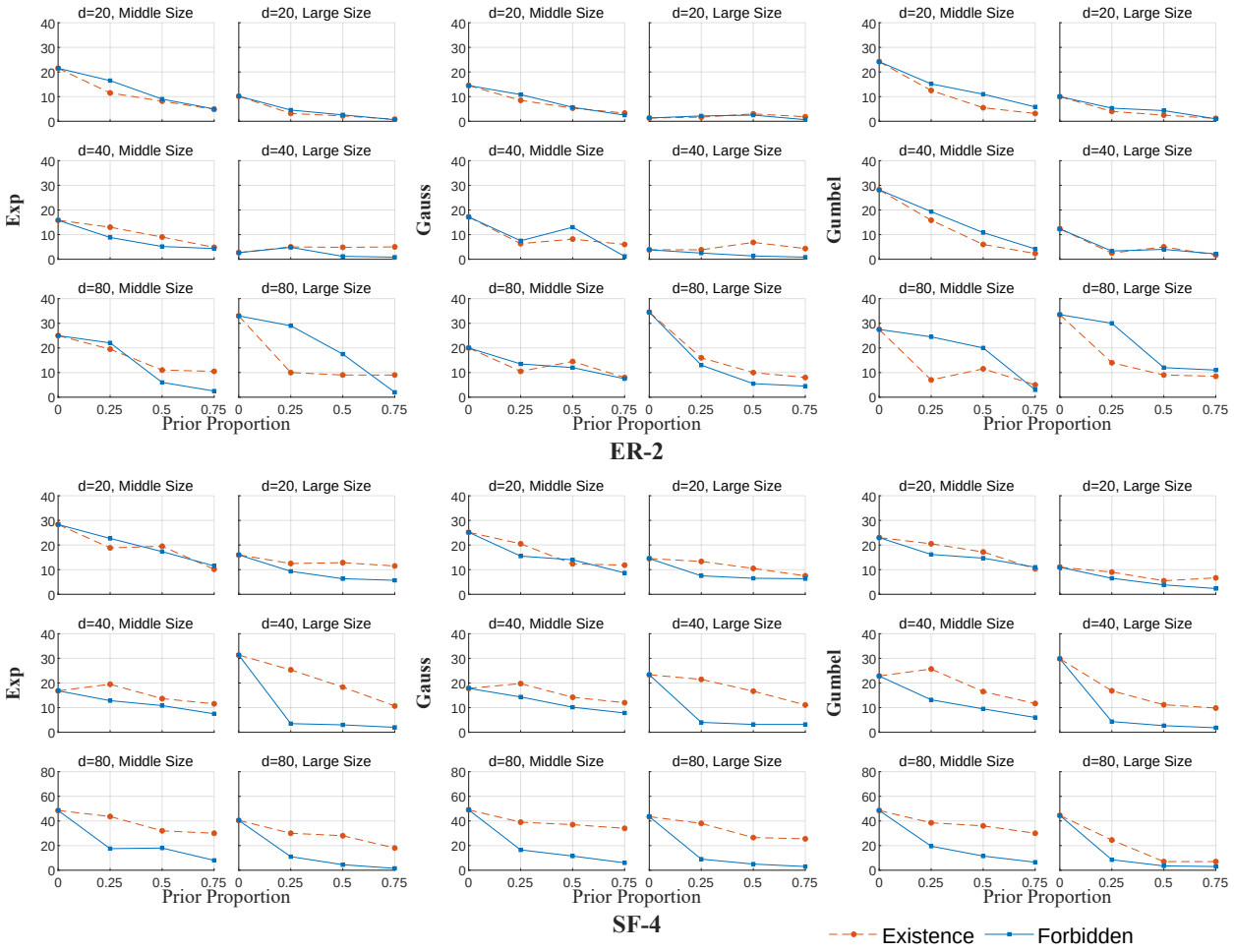

Figure 7: Performance of incorporating prior constraints in causal discovery on linear data. The y-axis represents the SHD of the method.

across a range of evaluation metrics.

A.4 Experiment on Non-Equal Variance Noise

In this experiment, we investigate the performance of our proposed method under conditions involving non-equal variance (NV) noise, a common challenge in causal discovery. Building on suggestions from Reisach et al., who posited that the NOTEARS method might benefit from the equivariance of noise in simulated data, we aim to demonstrate that our approach can effectively enhance the performance of NOTEARS in non-equal variance scenarios. To introduce NV noise, we scale each noise component of the data by a random variable uniformly distributed between 0.5 and 2, thereby creating data with heteroscedasticity. We also compare our method to the VARSORT method to assess its relative performance. The Table 4 shows the results.

The results clearly show that our method significantly improves the performance of the NOTEARS method when applied to data with NV noise. Under these conditions, the NOTEARS method tends to struggle with accurately determining the order of nodes, often resulting in erroneous causal structures. In contrast, our method leverages prior knowledge to appropriately adjust the node order, improving the overall structural accuracy, especially in nonlinear SEMs. By incorporating such priors, our method ensures that the causal relationships are better captured, even in the presence of NV noise.

Additionally, our method substantially outperforms the NOTEARS method in both the F1 score and SHD metrics, confirming its effectiveness in recovering the true graph structure more accurately under these challenging conditions. In terms of SID, while VARSORT still exhibits slightly superior performance, the smaller gap between our method and VARSORT indicates that our approach also mitigates the negative impact of NV noise. The fact that our method performs competitively with VARSORT suggests that it is highly robust to noise variability, making it a promising approach for real-world applications where noise heterogeneity is common.

Table 2: The Performance of Proposed ECA Method on Denser Graph

DataSize	Noise Type	Node	80											
		Graph Type	ER						SF					
		Graph Density	0.1			0.2			0.1			0.2		
		Prior Proportion	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID
2d	Exp	0	43.9	270.0	4662.7	10.3	685.3	5906.7	48.2	229.0	869.0	21.6	573.7	1917.0
		0.25	64.1	180.3	4141.3	41.7	558.0	5346.0	65.2	170.3	589.7	58.4	374.3	1108.0
		0.5	77.6	124.3	3255.3	63.2	408.3	4852.7	77.5	119.7	507.7	73.5	270.3	925.7
		0.75	**89.0**	**65.0**	**2291.3**	**78.8**	**265.3**	**3978.3**	**86.8**	**75.3**	**387.3**	**85.2**	**164.3**	**774.7**
	Gauss	0	45.7	262.0	4557.7	10.0	684.7	5890.0	48.1	229.3	926.0	20.3	584.3	2053.0
		0.25	65.4	176.3	3990.3	41.3	558.7	5241.7	64.3	174.0	563.3	57.8	382.3	1106.3
		0.5	76.0	132.0	3480.3	63.7	402.7	4823.0	78.1	118.0	485.0	72.9	277.0	905.3
		0.75	**88.6**	**66.7**	**2372.7**	**79.7**	**252.0**	**3958.0**	**87.4**	**73.0**	**356.7**	**85.4**	**163.3**	**765.7**
	Gumbel	0	41.5	284.0	4768.7	11.6	683.7	5917.0	44.5	252.7	969.3	20.1	596.0	2267.3
		0.25	65.1	177.3	4045.7	42.2	554.7	5193.7	65.3	171.0	574.7	57.4	385.3	1095.7
		0.5	77.1	125.7	3493.0	63.6	405.7	4786.0	77.5	122.3	463.0	72.3	283.0	895.3
		0.75	**90.0**	**58.7**	**2070.0**	**79.7**	**251.0**	**3758.3**	**87.6**	**71.3**	**361.7**	**84.3**	**174.3**	**784.0**
4d	Exp	0	39.0	292.0	4909.7	10.2	680.0	5978.7	45.2	243.7	906.3	20.5	593.0	2106.3
		0.25	64.3	181.0	4161.0	43.0	542.0	5277.7	65.7	168.3	563.3	57.5	381.0	1083.0
		0.5	76.8	126.0	3483.0	64.0	399.3	4829.7	77.6	119.7	506.3	72.9	274.3	909.3
		0.75	**89.8**	**60.0**	**2153.0**	**79.1**	**260.7**	**3893.3**	**87.4**	**72.0**	**379.7**	**84.6**	**169.3**	**772.3**
	Gauss	0	40.2	292.7	4662.3	10.8	681.7	5928.7	48.3	229.3	871.7	19.8	587.3	2117.7
		0.25	64.7	179.7	4050.7	42.4	555.3	5190.0	65.7	167.7	563.0	58.2	373.0	1132.0
		0.5	77.2	124.3	3427.3	63.9	401.7	4827.7	78.4	116.3	494.0	72.8	273.7	932.3
		0.75	**89.7**	**60.3**	**2251.7**	**79.3**	**255.7**	**3963.0**	**87.2**	**73.3**	**386.3**	**84.9**	**165.3**	**786.0**
	Gumbel	0	38.1	308.7	4696.7	11.0	678.3	5919.0	43.8	253.7	986.7	21.8	589.7	2226.7
		0.25	65.5	174.3	4075.3	42.8	544.3	5240.0	67.0	162.7	537.3	59.2	369.0	1100.3
		0.5	79.1	114.7	3397.0	64.9	392.7	4632.7	79.2	112.0	456.3	72.0	285.7	913.7
		0.75	**90.3**	**57.3**	**1971.0**	**79.5**	**254.3**	**3826.7**	**87.8**	**70.3**	**363.7**	**84.8**	**168.3**	**775.3**

Table 3: Comparison of Performance Between Traditional Methods and Proposed Method with Varying Noise Types.

Noise Type	Node	20						40					
	Graph Type	ER			SF			ER			SF		
	Metric	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID
Exp	Astar	62.3	21.8	178.3	66.0	36.0	143.2	63.1	42.8	495.2	64.8	76.3	541.2
	HC	67.2	17.8	170.3	66.0	35.0	137.2	65.3	41.2	542.5	65.8	74.5	515.8
	MINOBSx	64.2	20.5	173.0	66.1	35.8	140.3	66.5	40.0	476.3	64.6	76.2	531.5
	MMHC	67.6	18.5	147.2	66.6	34.0	120.2	70.0	35.8	458.0	66.9	72.3	416.2
	PC	53.9	26.7	209.5	51.5	42.3	181.8	52.2	51.0	680.3	52.6	89.2	653.2
	VARSORT	79.7	19.8	27.7	77.7	36.0	**32.0**	82.0	35.0	**21.7**	79.3	74.0	**65.0**
	ECA	**93.4**	**5.3**	**24.3**	**91.3**	**12.2**	32.8	**95.1**	**8.0**	26.5	**89.7**	**31.0**	103.7
Gauss	Astar	62.7	22.3	184.3	66.0	36.0	143.2	63.4	42.3	488.7	64.1	77.8	542.2
	HC	62.6	21.5	184.2	66.0	35.0	137.2	65.7	40.5	511.5	64.5	77.2	534.8
	MINOBSx	63.4	21.8	181.8	66.0	36.0	143.2	66.3	39.7	463.3	64.0	77.7	532.2
	MMHC	64.8	19.8	156.3	66.8	34.0	117.2	70.4	35.5	432.3	66.1	73.5	427.5
	PC	52.9	27.5	211.2	47.9	44.0	214.7	49.7	53.8	728.0	52.0	88.7	659.5
	VARSORT	74.0	27.2	29.5	78.6	34.2	**35.3**	76.0	50.0	35.3	80.5	68.2	**70.5**
	ECA	**95.0**	**3.8**	**15.8**	**84.9**	**19.5**	49.2	**95.8**	**6.8**	**25.8**	**89.5**	**31.8**	120.3
Gumbel	Astar	63.2	21.5	176.0	66.1	36.0	140.0	63.2	42.8	486.7	64.8	76.7	510.3
	HC	65.7	18.5	173.5	66.1	35.0	134.0	66.0	40.3	510.7	65.7	75.3	500.0
	MINOBSx	63.9	20.5	173.7	66.3	35.7	134.3	66.3	40.0	470.0	64.3	77.2	509.7
	MMHC	66.7	19.0	153.0	66.9	34.0	114.0	70.8	35.2	426.3	66.3	73.3	429.0
	PC	52.2	27.2	216.5	49.0	43.5	200.0	48.2	53.7	784.8	52.2	88.2	683.0
	VARSORT	77.2	23.3	29.3	82.3	28.3	**30.5**	76.3	49.7	38.0	79.2	73.8	**61.7**
	ECA	**92.7**	**5.8**	**19.8**	**89.0**	**14.8**	33.2	**96.4**	**5.8**	**26.0**	**92.3**	**23.0**	110.2

Table 4: Performance Comparison of Different Methods on Non-Equal Variance Noise in Linear and Nonlinear SEMs.

SEM Type	Noise Type	Node	20						40					
		Graph Type	ER			SF			ER			SF		
		Metric	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID	F1	SHD	SID
Linear	Exp-NV	VARSORT	68.6	29.5	45.7	77.0	32.7	53.3	76.9	43.2	97.7	75.9	82.0	83.7
		NOTEARS	65.1	34.8	53.7	78.5	31.5	47.2	70.4	56.5	126.2	83.5	54.2	78.5
		VARSORT+prior	73.4	27.0	39.5	84.0	24.7	**38.3**	80.3	39.2	**46.0**	79.8	73.5	**59.5**
		ECA	**86.7**	**11.2**	**34.3**	**88.5**	**16.2**	40.3	**91.3**	**14.0**	59.7	**89.8**	**31.8**	105.2
	Gauss-NV	VARSORT	71.2	25.8	59.3	71.9	40.3	49.8	74.3	49.3	92.2	72.9	90.8	119.5
		NOTEARS	60.7	36.8	72.0	77.9	31.3	41.8	70.8	54.7	127.8	85.5	45.5	98.2
		VARSORT+prior	76.2	23.7	45.8	78.6	33.3	**38.0**	77.0	46.3	64.5	77.0	82.7	101.5
		ECA	**87.5**	**10.3**	**25.5**	**86.2**	**18.7**	49.2	**91.0**	**14.5**	**61.8**	**91.8**	**24.7**	**80.3**
	Gumbel-NV	VARSORT	73.6	23.2	49.8	75.7	36.3	45.7	73.8	50.2	97.8	68.3	109.8	110.8
		NOTEARS	56.6	47.0	62.2	77.4	36.2	33.8	64.2	79.3	87.7	79.2	71.5	98.3
		VARSORT+prior	78.0	20.8	36.0	81.5	30.2	33.8	77.0	46.3	64.3	74.2	97.3	89.0
		ECA	**82.3**	**16.3**	**18.2**	**87.0**	**19.3**	**22.8**	**87.7**	**21.2**	**48.3**	**91.8**	**25.3**	**76.3**
Nonlinear	Mim-NV	VARSORT	55.0	29.0	169.3	58.5	40.5	128.5	60.4	55.8	480.8	58.4	93.0	463.2
		NOTEARS	52.3	26.0	173.7	20.8	61.8	140.3	41.6	59.0	495.8	15.1	138.5	423.0
		VARSORT+prior	68.0	25.7	154.2	**77.0**	**27.3**	112.2	70.5	49.3	450.2	**74.3**	**69.3**	426.5
		ECA	**78.8**	**14.3**	**129.5**	72.0	30.8	**103.2**	**76.9**	**30.5**	**329.7**	68.1	73.8	**277.3**
	MLP-NV	VARSORT	45.6	46.3	162.5	52.4	52.7	123.0	42.9	105.7	514.5	50.2	130.8	452.8
		NOTEARS	56.3	28.5	146.2	58.5	41.8	109.8	52.6	56.0	448.7	43.7	112.3	373.5
		VARSORT+prior	59.5	40.3	146.2	70.8	38.8	107.7	55.6	94.7	477.7	66.8	102.3	404.5
		ECA	**73.3**	**20.7**	**100.8**	**76.4**	**28.0**	**82.0**	**75.0**	**35.5**	**310.5**	**69.9**	**74.0**	**269.7**