Sparse Additive Model Pruning for Order-Based Causal Structure Learning

Kentaro Kanamori, Hirofumi Suzuki, Takuya Takagi

Artificial Intelligence Laboratory, Fujitsu Limited
{k.kanamori, suzuki-hirofumi, takagi.takuya}@fujitsu.com

Abstract

Causal structure learning, also known as causal discovery, aims to estimate causal relationships between variables as a form of a causal directed acyclic graph (DAG) from observational data. One of the major frameworks is the order-based approach that first estimates a topological order of the underlying DAG and then prunes spurious edges from the fully-connected DAG induced by the estimated topological order. Previous studies often focus on the former ordering step because it can dramatically reduce the search space of DAGs. In practice, the latter pruning step is equally crucial for ensuring both computational efficiency and estimation accuracy. Most existing methods employ a pruning technique based on generalized additive models and hypothesis testing, commonly known as CAM-pruning. However, this approach can be a computational bottleneck as it requires repeatedly fitting additive models for all variables. Furthermore, it may harm estimation quality due to multiple testing. To address these issues, we introduce a new pruning method based on sparse additive models, which enables direct pruning of redundant edges without relying on hypothesis testing. We propose an efficient algorithm for learning sparse additive models by combining the randomized tree embedding technique with group-wise sparse regression. Experimental results on both synthetic and real datasets demonstrated that our method is significantly faster than existing pruning methods while maintaining comparable or superior accuracy.

1 Introduction

In several scientific fields, such as biology and economics, it is often important to identify causal relationships between variables in observational data. *Causal structure learning*, also known as *causal discovery*, aims to estimate them as a form of a *causal directed acyclic graph (DAG)* (Pearl 2009). A causal DAG enables us to understand the relationships between variables and to predict the effect of interventions on the variables, which are crucial for decision-making in various applications (Peters, Janzing, and Schölkopf 2017).

One of the promising algorithmic frameworks for causal structure learning is the *order-based approach* (Teyssier and Koller 2005). Under some conditions on the data-generating process, previous studies have shown that the underlying causal DAG is identifiable from purely observational data (Peters et al. 2014). However, we need to search for a causal graph while ensuring it is acyclic, which incurs super-exponential computational costs in the number of variables (Chickering 1996). To alleviate this issue, the order-based approach first estimates a topological order of the underlying causal DAG and then prunes spurious edges from the fully-connected DAG induced by the estimated topological order, as illustrated in Figure 1. By estimating a topological order in advance, we can obtain a causal DAG without explicitly imposing the acyclicity constraint, which reduces the search space dramatically (Rolland et al. 2022).

While existing studies often focus on the former ordering step, in practice, the latter pruning step is equally crucial for ensuring both computational efficiency and estimation accuracy. Most existing methods employ a pruning algorithm based on generalized additive models (GAMs) (Hastie and Tibshirani 1986), known as *CAM-pruning* (Bühlmann, Peters, and Ernest 2014). It first fits a GAM for regressing each variable on its candidate parents, and then identifies redundant parents by hypothesis testing for the fitted GAM (Marra and Wood 2011). However, the computational cost of fitting GAMs is generally expensive, especially for high-dimensional cases. Since CAM-pruning needs to repeatedly fit GAMs for all variables, it often becomes the bottleneck of the entire algorithm (Rolland et al. 2022; Montagna et al. 2023). Furthermore, CAM-pruning also requires repeating hypothesis testing, which can degrade the estimation accuracy due to multiple testing (Huang et al. 2018).

The goal of this paper is to propose an alternative pruning method that addresses the aforementioned limitations of existing pruning methods. It enables us to accelerate the existing order-based causal structure learning algorithms without compromising their estimation quality.

Our Contributions In this paper, we propose a new pruning method for order-based causal structure learning. Our key idea is to learn a *sparse additive model* (Ravikumar et al. 2009) that regresses each variable on its candidate parents, which enables us to directly prune redundant candidate parents without requiring hypothesis testing. To accelerate the process of fitting sparse additive models, we propose a new framework, named *Sparse Additive Randomized TRee Ensemble (SARTRE)*, by combining the randomized tree embedding and group-wise sparse regression techniques. Our

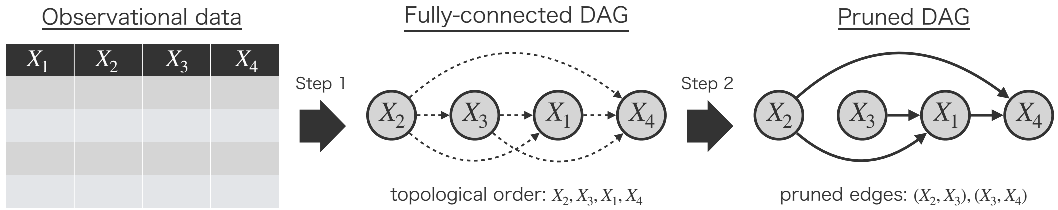


Figure 1: An overview of the order-based causal structure learning algorithm. Given an observational dataset, it first estimates a topological order of the underlying causal DAG. Then, it prunes spurious edges from the fully-connected DAG induced by the estimated topological order. This paper focuses on the latter step and aims to propose an efficient and accurate pruning method.

contributions are summarized as follows:

- We introduce a new efficient framework for learning a sparse additive model, named SARTRE. We consider a special case of the additive model, where the shape function for each variable is expressed as a linear combination of weighted indicator functions over a set of intervals. We propose to generate a set of intervals by *randomized tree embedding* (Moosmann, Triggs, and Jurie 2006), and show that we can efficiently learn the sparse weight vector via *group lasso regression* (Yuan and Lin 2006).
- We propose an efficient pruning method for order-based causal structure learning by leveraging our SARTRE framework. Given an estimated topological order, our method can efficiently prune redundant edges from the fully-connected DAG induced by the estimated topological order without requiring hypothesis testing. Our method can be combined with any causal ordering algorithm, such as SCORE (Rolland et al. 2022).
- By numerical experiments on synthetic and real datasets, we demonstrated that our method achieved a significant speedup compared to existing pruning methods, including CAM-pruning (Bühlmann, Peters, and Ernest 2014). Furthermore, our method achieved comparable or superior accuracy to existing methods, suggesting that it can be a promising alternative to current pruning methods.

2 Preliminaries

2.1 Causal Structure Learning

Causal structure learning, also known as *causal discovery*, aims to estimate a *causal graph* that expresses the causal relationships between variables from observational data. For a set of variables $[d] := \{1, \dots, d\}$, we consider an *additive noise model (ANM)* with a directed acyclic graph (DAG) $\mathcal{G}$ (Peters et al. 2014). More precisely, we assume that a random variable $X = (X_1, \dots, X_d) \in \mathbb{R}^d$ is generated by the following *structural equation model* for each $i \in [d]$:

$$X_i = f_i(X_{\mathrm{pa}(i)}) + \varepsilon_i,$$

where $X_{\mathrm{pa}(i)}$ is a vector of variables that are parents of X_i in $\mathcal{G}$, f_i is a deterministic link function, and $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$ is an independent Gaussian noise variable.

As with the previous studies (Peters et al. 2014; Rolland et al. 2022), we assume that the link functions f_i are nonlinear and twice continuously differentiable in every component. Under mild conditions, the ANM defined above is known to be *identifiable*; that is, the underlying causal DAG $\mathcal{G}$ can be uniquely recovered from observational data generated according to the joint distribution of X (Peters et al. 2014). Motivated by this fact, several algorithms for estimating the causal DAG $\mathcal{G}$ from observational data have been proposed so far (Bühlmann, Peters, and Ernest 2014; Montagna et al. 2023; Xu et al. 2024).

2.2 Order-Based Causal Structure Learning

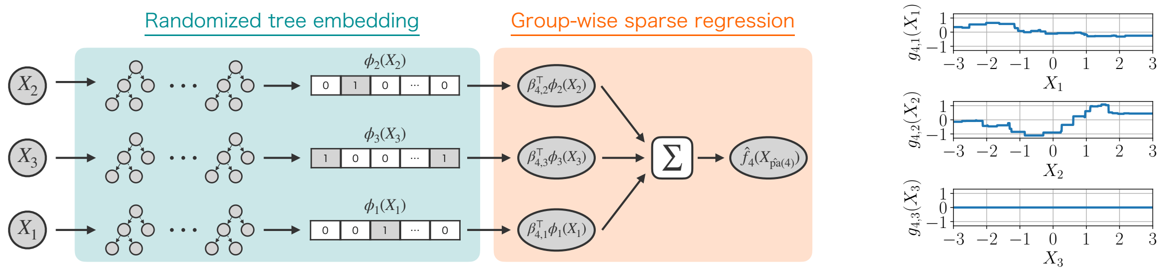
One of the major frameworks for estimating a causal DAG $\mathcal{G}$ is the *order-based approach* (Teyssier and Koller 2005). Given an observational sample $S = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\} \subseteq \mathbb{R}^d$, it divides the task of estimating $\mathcal{G}$ into two steps: (1) estimating a topological order of $\mathcal{G}$, and (2) pruning redundant edges from the fully-connected DAG induced by the estimated topological order. Figure 1 presents an overview of the order-based approach. Previous studies often focus on the former step and proposed several methods for estimating a topological order from a given observational sample S. For the latter pruning step, most of the existing methods employ a nonlinear variable selection algorithm based on *generalized additive models (GAMs)* (Hastie and Tibshirani 1986).

Ordering Step Given a sample S, the ordering step aims to estimate a topological order π of the underlying DAG $\mathcal{G}$. A topological order π is expressed as a permutation over $[d]$ such that, for any $i, j \in [d]$, $\pi(i) < \pi(j)$ holds if X_j is a descendant of X_i in $\mathcal{G}$. Existing methods often estimate a topological order in a bottom-up greedy manner that iteratively identifies a leaf variable, i.e., a variable that is not a parent of any other remaining variables (Peters et al. 2014).

One of the state-of-the-art methods is the *SCORE* algorithm (Rolland et al. 2022), which identifies a leaf variable by leveraging the score matching technique. For the underlying joint distribution $p(x)$ of X, let $s(x) := \nabla \log p(x)$ be the logarithmic gradient of $p(x)$, which is known as the *score function*. Then, (Rolland et al. 2022) proved that X_i is a leaf variable if and only if the variance of $\frac{\partial s_i(X)}{\partial x_i}$ is zero. Based on this fact, SCORE identifies a leaf variable X_l by

$$l = \arg\min_{i \in [d]} \mathrm{Var}_X \left[\frac{\partial s_i(X)}{\partial x_i} \right].$$

Note that $\mathrm{Var}_X \left[\frac{\partial s_i(X)}{\partial x_i} \right]$ can be estimated by the second-order Stein gradient estimator (Li and Turner 2018) with a given sample S. After identifying a leaf variable X_l, SCORE



(a) Model Overview (b) Shape Functions

Figure 2: An example of our SARTRE framework. We consider the same example as in Figure 1, where the estimated topological order $\hat{\pi} = (2, 3, 1, 4)$ is given and we consider to identify the parents of the variable X_4 from its candidate parents $\hat{\mathrm{pa}}(4) = \{2, 3, 1\}$. (a) Our method first generates binary representation vectors $(\phi_2(X_2), \phi_3(X_3), \phi_1(X_1))$ by randomized tree embedding. Then, it learns an additive model $\hat{f}_4(X_{\hat{\mathrm{pa}}(4)}) = g_{4,2}(X_2) + g_{4,3}(X_3) + g_{4,1}(X_1)$, where each shape function is defined by $g_{4,j}(X_j) = \boldsymbol{\beta}_{4,j}^{\top}\phi_j(X_j)$. By optimizing the coefficient vectors $(\boldsymbol{\beta}_{4,2}, \boldsymbol{\beta}_{4,3}, \boldsymbol{\beta}_{4,1})$ through group lasso regression, we can obtain a sparse additive model $\hat{f}_4$. (b) For each shape function $g_{4,j}$, if $\boldsymbol{\beta}_{4,j} = \mathbf{0}$ holds, we have $g_{4,j}(X_j) = 0$ for any X_j, which enables us to prune the corresponding candidate parent X_j. In this example, $\boldsymbol{\beta}_{4,3} = \mathbf{0}$ holds and thus we can prune X_3.

removes it from the set of variables and repeats this procedure for the remaining variables. When all variables are removed, we can obtain an estimated topological order $\hat{\pi}$.

Pruning Step Given an estimated topological order $\hat{\pi}$ and S, the pruning step aims to remove spurious edges and identify the true parents of each variable X_i in the underlying DAG $\mathcal{G}$. Once a topological order $\hat{\pi}$ is determined, we can construct the fully-connected DAG by adding a directed edge from X_j to X_i for each $j, i \in [d]$ such that $\hat{\pi}(j) < \hat{\pi}(i)$ holds. Let $\hat{\mathrm{pa}}(i) := \{j \in [d] \mid \hat{\pi}(j) < \hat{\pi}(i)\}$ be the set of *candidate parents* of X_i with respect to $\hat{\pi}$. Then, the task of the pruning step is to select the true parents of each X_i from $\hat{\mathrm{pa}}(i)$, which can be regarded as a variable selection problem. Most existing methods employ a variable selection method based on GAMs, known as *CAM-pruning* (Bühlmann, Peters, and Ernest 2014). The idea behind CAM-pruning is to assume that the link function f_i can be expressed as an additive model. It first fits an additive model that regresses X_i on its candidate parents $X_{\hat{\mathrm{pa}}(i)}$:

$$\hat{f}_i(X_{\hat{\mathrm{pa}}(i)}) = \sum\nolimits_{j \in \hat{\mathrm{pa}}(i)} g_{i,j}(X_j),$$

where $g_{i,j}$ is a nonlinear *shape function* for each $j \in \hat{\mathrm{pa}}(i)$. Then, CAM-pruning selects a subset of $\hat{\mathrm{pa}}(i)$ by hypothesis testing whether $g_{i,j}$ is significantly different from zero for each $j \in \hat{\mathrm{pa}}(i)$. That is, it removes each candidate parent X_j if the null hypothesis $g_{i,j}(X_j) = 0$ is accepted.

While CAM-pruning is known as a powerful method, it faces two critical challenges. First, while it needs to fit an additive model for each variable X_i, repeating this procedure for all variables can be computationally expensive, especially in high-dimensional cases (Montagna et al. 2023). Second, it requires hypothesis testing for all pairs of each variable and its candidate parent, which can harm the correctness of pruning due to multiple testing (Huang et al. 2018). To address these issues, the aim of this paper is to propose an alternative pruning method that can be applied to high-dimensional datasets while avoiding multiple testing.

3 Algorithm

In this section, we propose an efficient and accurate pruning method for order-based causal structure learning. We assume that we are given a topological order $\hat{\pi}$ over variables $[d]$ estimated from an observational sample $S = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\} \subseteq \mathbb{R}^d$ by some algorithm. Note that our method can be combined with any existing ordering algorithm, including CAM (Bühlmann, Peters, and Ernest 2014), SCORE (Rolland et al. 2022), and CaPS (Xu et al. 2024). Given an estimated topological order $\hat{\pi}$, we aim to identify the true parents $\mathrm{pa}(i)$ for each variable X_i by removing spurious variables from its candidate parents $\hat{\mathrm{pa}}(i)$ given by $\hat{\pi}$.

Our main idea is to learn a *sparse additive model* (Ravikumar et al. 2009) that regresses each variable X_i on a small subset of its candidate parents $X_{\hat{\mathrm{pa}}(i)}$. By learning a sparse additive model, we can directly prune redundant candidate parents without relying on hypothesis testing. However, fitting a sparse additive model for each variable X_i can be computationally expensive, as is the case with CAM-pruning. Since we need to repeat this procedure for all variables, the computational cost can be prohibitive in practice. To address this issue, we propose a new framework for efficiently learning sparse additive models by combining the *randomized tree embedding* (Moosmann, Triggs, and Jurie 2006) and *group lasso regression* (Yuan and Lin 2006).

3.1 Sparse Additive Randomized Tree Ensemble

First, we introduce *Sparse Additive Randomized TRee Ensemble (SARTRE)*, a new framework for learning sparse additive models. We consider a special case of the additive model, where each shape function is expressed as a linear combination of weighted indicator functions over a set of intervals. More precisely, we consider an additive model $\hat{f}_i(X_{\hat{\mathrm{pa}}(i)}) = \sum_{j \in \hat{\mathrm{pa}}(i)} g_{i,j}(X_j)$ whose shape functions $g_{i,j}$ are defined as follows:

$$g_{i,j}(X_j) = \sum\nolimits_{k=1}^{l_j} \beta_{i,j,k} \cdot \phi_{j,k}(X_j),$$

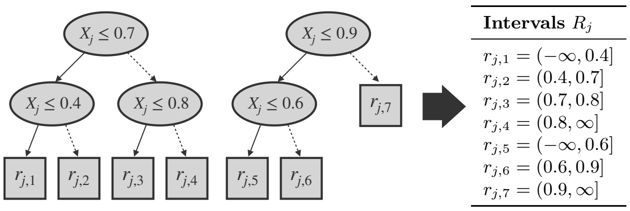


Figure 3: An example of the tree embedding technique. Given an ensemble of decision trees that takes X_j as input, each leaf k of a tree corresponds to an interval $r_{j,k}$ of X_j. Thus, we can obtain a set of intervals R_j by collecting intervals corresponding to the leaves in a given tree ensemble.

Algorithm 1: SARTRE-pruning

```
for j = 1, ..., d do                                   ▷ Randomized Tree Embedding
    Fit a randomized tree ensemble h_j;
    Extract R_j = {r_{j,1}, ..., r_{j,l_j}} from h_j;
end for
Construct a fully-connected DAG Ĝ induced by π̂.
for i = 1, ..., d do                                   ▷ Group-wise Sparse Regression
    β̂_i ← argmin_{β_i ∈ R^{p_i}} L(β_i) + λ · Σ_{j ∈ pâ(i)} ||β_{i,j}||_2;
    for j ∈ pâ(i) do
        if β̂_{i,j,1} = ··· = β̂_{i,j,l_j} = 0 then
            Remove the edge (X_j, X_i) from Ĝ;
        end if
    end for
end for
return Ĝ
```

where l_j is the total number of intervals for X_j, $\phi_{j,k}(X_j) = \mathbb{I}[X_j \in r_{j,k}]$ is an indicator function with respect to the k-th interval $r_{j,k} \subset \mathbb{R}$, and $\beta_{i,j,k}$ is a coefficient for $r_{j,k}$. Each interval $r_{j,k}$ is expressed as $r_{j,k} = (a_{j,k}, b_{j,k}]$ with lower and upper bounds $a_{j,k}, b_{j,k} \in \mathbb{R}$. For notational convenience, we denote by $\phi_j(X_j) := (\phi_{j,1}(X_j), \dots, \phi_{j,l_j}(X_j)) \in \{0,1\}^{l_j}$, which can be regarded as an embedding vector of X_j using the set of intervals $R_j := \{r_{j,1}, \dots, r_{j,l_j}\}$. Then, our shape function can be expressed by $g_{i,j}(X_j) = \boldsymbol{\beta}_{i,j}^\top \phi_j(X_j)$, where $\boldsymbol{\beta}_{i,j} := (\beta_{i,j,1}, \dots, \beta_{i,j,l_j}) \in \mathbb{R}^{l_j}$ is a coefficient vector for the pair of a variable X_i and its candidate parent X_j.

To learn our additive model $\hat{f}_i$, we need to generate a set of intervals R_j and optimize the coefficient vector $\boldsymbol{\beta}_{i,j}$ for each $j \in \hat{\text{pa}}(i)$. By definition, it is easy to see that $g_{i,j}(X_j) = 0$ if $\beta_{i,j,k} = 0$ holds for all $k \in [l_j]$. It implies that we can obtain a sparse additive model by forcing $\boldsymbol{\beta}_{i,j} = \mathbf{0}$ for as many candidate parents $j \in \hat{\text{pa}}(i)$ as possible. Based on this observation, our SARTRE consists of the following two steps: (1) *randomized tree embedding*, which efficiently generates a set of intervals R_j by an unsupervised manner; (2) *group-wise sparse regression*, which optimizes each coefficient vector $\boldsymbol{\beta}_{i,j}$ by group lasso regression. Figure 2 illustrates an overview of our SARTRE framework.

Randomized Tree Embedding To generate a set of intervals R_j for each $j \in \hat{\text{pa}}(i)$, we employ the embedding technique based on tree ensemble models (Moosmann, Triggs, and Jurie 2006). A tree ensemble model consists of decision trees that partition the input space and assign a constant value to each partition. Given a tree ensemble $h_j : \mathbb{R} \to \mathbb{R}$ that takes one single variable X_j as input, each leaf k of a decision tree in the ensemble corresponds to an interval $r_{j,k}$. It suggests that we can obtain R_j by collecting intervals corresponding to the leaves in a given tree ensemble h_j. Such a technique is known as *tree embedding* (Moosmann, Triggs, and Jurie 2006), which enables us to easily express nonlinear relationships (Friedman and Popescu 2008; Feng and Zhou 2018). Figure 3 illustrates a running example of how to extract a set of intervals R_j from a tree ensemble h_j.

There exist several possible ways to obtain a tree ensemble h_j. One straightforward approach is to train a tree ensemble h_j that regresses X_i on X_j by some supervised learning algorithm, such as random forests (Breiman 2001) or gradient boosting (Friedman 2000). To accelerate our generating process, we employ an ensemble of completely randomized trees (Geurts, Ernst, and Wehenkel 2006). Because these trees are constructed by randomly selecting a split point for each node without any target variable, we can generate R_j more efficiently than the supervised approach. Note that the intervals R_j that are generated in such an unsupervised and randomized manner can not always be optimal in terms of the regression performance. However, since we optimize the corresponding coefficient vectors $\boldsymbol{\beta}_{i,j}$ in the next step, they may not harm the overall performance if we can set the total number of intervals to be sufficiently large (Geurts, Ernst, and Wehenkel 2006).

Group-Wise Sparse Regression Given the generated intervals R_j for each $j \in \hat{\text{pa}}(i)$, we optimize the corresponding coefficient vectors $\boldsymbol{\beta}_{i,j}$ by the group-wise sparse regression technique (Yuan and Lin 2006; Massias, Gramfort, and Salmon 2018). Let $p_i = \sum_{j \in \hat{\text{pa}}(i)} l_j$ be the total number of intervals for all candidate parents of X_i. We denote the concatenated embedding and coefficient vectors by $\Phi_i(\boldsymbol{x}) := (\phi_j(x_j))_{j \in \hat{\text{pa}}(i)} \in \{0,1\}^{p_i}$ and $\boldsymbol{\beta}_i := (\boldsymbol{\beta}_{i,j})_{j \in \hat{\text{pa}}(i)} \in \mathbb{R}^{p_i}$, respectively. Then, our additive model can be expressed as $\hat{f}_i(\boldsymbol{x}_{\hat{\text{pa}}(i)}) = \boldsymbol{\beta}_i^\top \Phi_i(\boldsymbol{x})$. It implies that our additive model can be regarded as a linear model over embedding Φ_i with a coefficient vector $\boldsymbol{\beta}_i$ if the intervals R_j are fixed. In addition, our coefficient vector $\boldsymbol{\beta}_i$ has a group structure, where each group corresponds to one candidate parent of X_i, and we aim to encourage group-wise sparsity in $\boldsymbol{\beta}_i$ for obtaining a sparse additive model.

By leveraging the facts mentioned above, we can fit a coefficient vector $\boldsymbol{\beta}_i$ by *group lasso regression* (Yuan and Lin 2006). Our problem can be formulated as follows:

$$\min_{\boldsymbol{\beta}_i \in \mathbb{R}^{p_i}} \mathcal{L}(\boldsymbol{\beta}_i) + \lambda \cdot \sum_{j \in \hat{\text{pa}}(i)} \|\boldsymbol{\beta}_{i,j}\|_2,$$

where $\mathcal{L}(\boldsymbol{\beta}_i) := \sum_{m=1}^{n} \left(x_{m,i} - \boldsymbol{\beta}_i^\top \Phi_i(\boldsymbol{x}_m)\right)^2$ and $\lambda > 0$ is a regularization parameter. While the first term of the objective function is the standard squared loss, the second term is the group lasso penalty that encourages group-wise sparsity in $\boldsymbol{\beta}_i$. We can solve the above optimization problem by existing efficient algorithms, such as block-coordinate descent (Friedman, Hastie, and Tibshirani 2010) or dual extrapolation (Massias, Gramfort, and Salmon 2018).

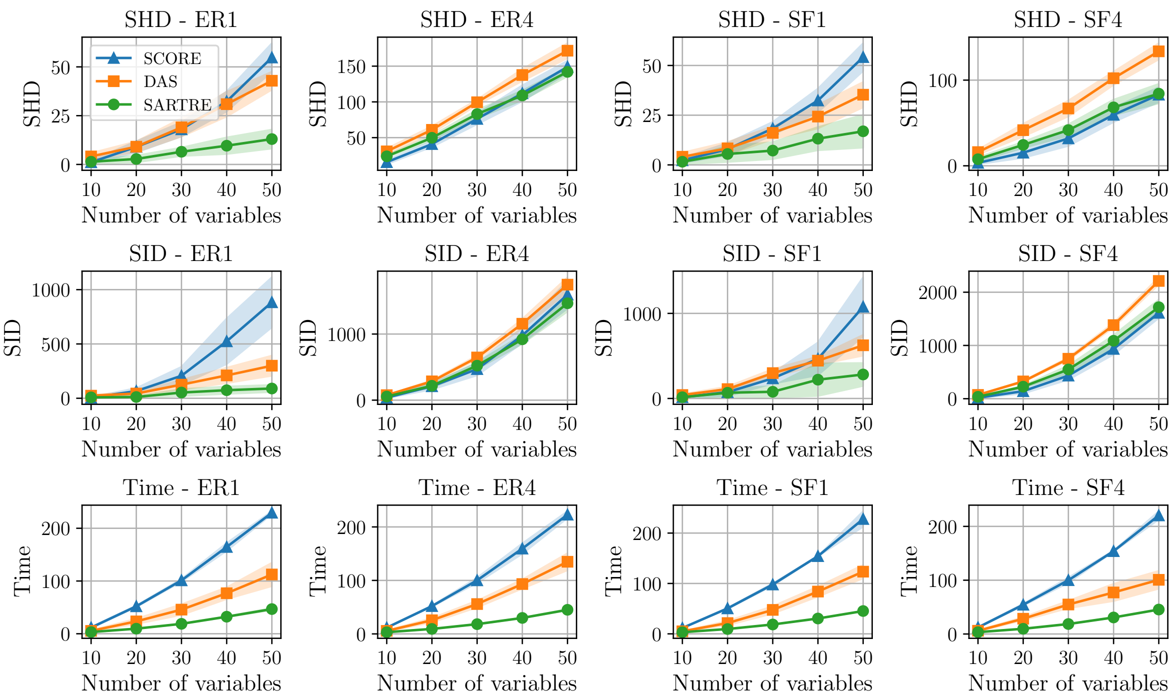


Figure 4: Experimental results of baseline comparison on the synthetic datasets. For all the metrics, smaller values are better. The shaded areas indicate the standard deviations over 10 trials. We varied the number of variables d from 10 to 50. Our SARTRE was significantly faster than the baselines while maintaining comparable or superior SHD and SID.

3.2 Overall Pruning Algorithm

We now propose a pruning algorithm for order-based causal structure learning, based on our SARTRE framework. Algorithm 1 presents our algorithm, named *SARTRE-pruning*. Given an estimated topological order $\hat{\pi}$ and a sample S, it first generates a set of intervals R_j for each variable X_j by randomized tree embedding. Then, for each variable X_i, it trains a sparse additive model $\hat{f}_i(X_{\hat{\mathrm{pa}}(i)})$ by optimizing the coefficient vector $\hat{\boldsymbol{\beta}}_i$ through group lasso regression. For a candidate parent $j \in \hat{\mathrm{pa}}(i)$, if all the coefficients $\hat{\beta}_{i,j,k}$ are zero, we conclude that X_j is not a parent of X_i and prune the edge (X_j, X_i) from the fully-connected DAG $\hat{\mathcal{G}}$ induced by $\hat{\pi}$. Finally, our algorithm returns the pruned DAG $\hat{\mathcal{G}}$.

Our SARTRE-pruning has several advantages compared to the existing pruning methods (Bühlmann, Peters, and Ernest 2014; Montagna et al. 2023). In terms of computational efficiency, our algorithm generates a set of intervals R_j for each variable in an unsupervised manner (lines 1–4) before the pruning procedure. Since the generated intervals R_j are independent of any target variable, we can use R_j for all the variables whose candidate parents include X_j. Thus, all we need in our pruning procedure is to optimize the coefficient vector $\hat{\boldsymbol{\beta}}_i$ for each X_i (lines 6–13), which is more efficient than fitting an additive model from scratch. Furthermore, our algorithm obtains a sparse additive model for each variable X_i that encourages as many shape functions $g_{i,j}$ as possible to be zero (Ravikumar et al. 2009). It enables us to directly identify spurious edges without multiple testing.

3.3 Theoretical Analysis

Finally, we discuss the representation ability of our additive model compared to the standard one used in CAM-pruning. As a shape function, existing implementations of CAM-pruning often employ a *smoothing spline* (Hastie, Tibshirani, and Friedman 2009), which is a smooth piece-wise polynomial function. On the other hand, our shape function $g_{i,j}$ does not include polynomial terms and only consists of a linear combination of weighted indicator functions over a set of intervals. Contrary to such a simple structure, we show that our shape function has the potential to express any continuous function arbitrarily well in Proposition 1.

Proposition 1. *For a variable X_j, we assume $X_j \in [a_j, b_j]$ for some $a_j < b_j$. Then, for any continuous function $g^* \colon [a_j, b_j] \to \mathbb{R}$ and $\varepsilon > 0$, there exist our shape function $g_{i,j}$ such that $\max_{x_j \in [a_j,b_j]} |g^*(x_j) - g_{i,j}(x_j)| < \varepsilon$ holds.*

Proof (sketch). By definition, our shape function $g_{i,j}$ can be expressed as a *piece-wise constant function*, which is a universal approximator for any continuous function (Cybenko 1989). This fact implies the existence of $g_{i,j}$ that can approximate a given continuous function g^* arbitrarily well. $\square$

Proposition 1 suggests that our additive model potentially has a rich representation ability, while it is restricted to a simple shape function compared to CAM-pruning. Unfortunately, it is not trivial to determine the sufficient number of intervals l_j and learn the appropriate coefficients $\boldsymbol{\beta}_{i,j}$. While we leave them for future work, in the next section, we empirically demonstrate that our method works well in practice.

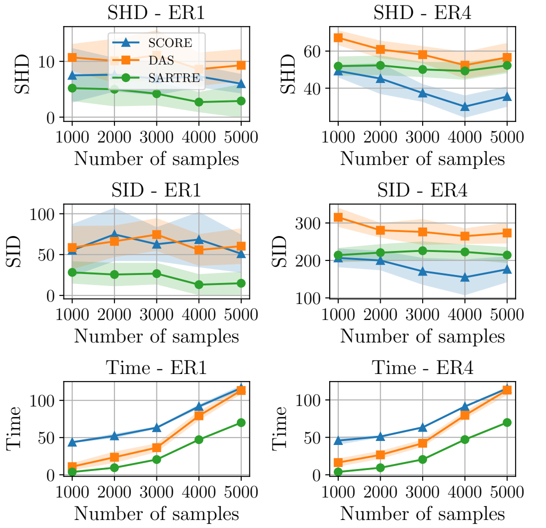


Figure 5: Experimental results of baseline comparison by varying the number of samples n from 1000 to 5000 on ER1 and ER4 datasets. Our SARTRE was faster than the baselines without significantly degrading SHD and SID.

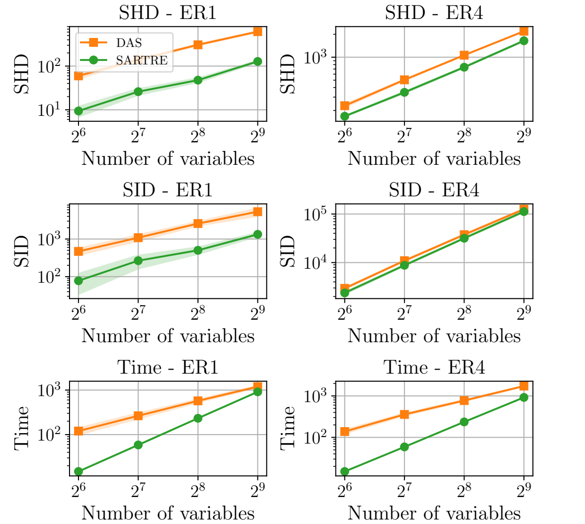


Figure 6: Experimental results of the high-dimensional cases on ER1 and ER4 datasets. We varied the number of variables by $d \in \{64, 128, 256, 512\}$. We observed that our SARTRE was faster and attained better SHD and SID than DAS.

4 Experiments

To investigate the efficacy of our framework, we conducted numerical experiments on synthetic and real datasets. All the code was implemented in Python 3.10. All the experiments were conducted on macOS Sequoia with Apple M2 Ultra CPU and 128 GB memory. Due to page limitations, the complete results are shown in Appendix.

4.1 Experimental Setup

Datasets We examine the performance of our method on synthetic datasets generated from a nonlinear ANM with Gaussian noise. Following (Rolland et al. 2022), we generated link functions f_i by sampling Gaussian processes with a unit bandwidth RBF kernel. We considered two types of causal DAGs: *Erdős–Rényi (ER)* and *scale-free (SF)* models. For a fixed number of variables d, we varied the sparsity of the sampled graph by setting the average number of edges to be d or $4d$. We also used real and semi-real datasets: *Sachs* (Sachs et al. 2005) and *fMRI* (Smith et al. 2011).

Baselines We compare our method (*SARTRE*) with two existing baselines. One baseline is *SCORE* (Rolland et al. 2022), which estimates a topological order by the score matching and then prunes spurious edges by CAM-pruning. Another baseline is *DAS* (Montagna et al. 2023), which is a fast variant of SCORE that filters redundant candidate parents before applying CAM-pruning. Some other methods, such as CAM (Bühlmann, Peters, and Ernest 2014), NOTEARS (Zheng et al. 2018, 2020), and GraNDAG (Lachapelle et al. 2020), are not included in our comparison since they were outperformed by SCORE in (Rolland et al. 2022). We implemented our SARTRE by combining the ordering step of SCORE and Algorithm 1. Note that all methods use the same ordering step, and their differences are only in the pruning step. For SARTRE, we set $\lambda = 0.1$ because it performed the best in our sensitivity analyses, which are shown in Appendix. We also set the total number of trees and the maximum leaf size of each tree in the ensemble h_j to be 5 and 8, respectively, which means that the total number of intervals l_j is at most 40. We repeated each experiment 10 times and report the average values of (i) structural Hamming distance (SHD), (ii) structural interventional distance (SID) (Peters and Bühlmann 2015), and (iii) running time [s] for each method.

4.2 Experimental Results

Baseline Comparison First, we evaluate the performance of each method by varying the number of variables d. Figure 4 shows the average SHD, SID, and running time of each method on the synthetic datasets with $n = 2000$. We can see that our SARTRE was significantly faster than the baselines, as the number of variables d approached 50. Furthermore, while SARTRE maintained comparable SHD and SID to the baselines on ER4 and SF4, it achieved the best SHD and SID on ER1 and SF1. It indicates that SARTRE was more accurate than the baselines when the underlying causal graph was sparse, while it remained competitive on dense graphs. In summary, we confirmed that *our SARTRE achieved significant speedup compared to the baselines while maintaining comparable or superior accuracy*.

We also compare each method by varying the number of samples n. Figure 5 presents the results on ER1 and ER4 datasets with $d = 20$. We observed that the running time of DAS approached that of SCORE as the number of samples n increased, while SARTRE remained faster than both baselines. In terms of accuracy, SCORE attained better SHD and SID than SARTRE on ER4. One possible reason is that hypothesis testing of CAM-pruning remains accurate for large

Method	Sachs SHD	Sachs SID	Sachs Time	fMRI SHD	fMRI SID	fMRI Time
SCORE	43.2	102.4	14.7	19.6	71.8	11.4
DAS	27.6	70.3	6.42	**11.6**	**58.6**	4.75
SARTRE	**22.7**	**58.0**	**3.62**	12.9	60.0	**3.18**

Table 1: Experimental results on real datasets. We sampled each dataset by bootstrap sampling, and repeated this procedure for 10 trials. We can see that our SARTRE was faster than the baselines while maintaining comparable accuracy.

n relatively to d and dense DAGs. Nevertheless, SARTRE outperformed SCORE on ER1 and DAS on both ER1 and ER4. These results indicate that *our SARTRE often performed well with respect to the number of samples as well*.

High-Dimensional Case Next, we examine our method in high-dimensional cases. We varied d by $\{64, 128, 256, 512\}$ and fixed $n = 2000$. Due to the high dimensionality, we omitted the ordering step of each method and used the ground truth topological order as input. Figure 6 shows the results on ER1 and ER4 datasets for DAS and SARTRE. We observed that SARTRE was faster than DAS while maintaining better SHD and SID in all cases. From these results, we confirmed that *our SARTRE performed better than the existing scalable algorithm even in the high-dimensional cases*.

Real Datasets Finally, we evaluate our method on real and semi-real datasets. We sampled a subset of each dataset by bootstrap sampling with $n = 2000$. Table 1 shows the average SHD, SID, and running time of each method on Sachs and fMRI datasets over 10 trials. We observed that SARTRE was faster than the baselines without significantly degrading SHD and SID. These results suggest that *our SARTRE performed well on real datasets, as well as synthetic datasets*.

5 Related Work

Causal structure learning, also referred to as *causal discovery*, has been a fundamental task in the field of statistics, machine learning, and artificial intelligence for decades (Pearl 2009; Peters, Janzing, and Schölkopf 2017). Existing studies can be categorized into three main approaches and mixtures of them: *constraint-based* (Spirtes and Glymour 1991), *score-based* (Chickering 2003), and *functional model-based methods* (Shimizu et al. 2006). This paper focuses on functional model-based methods, and in particular, considers the *additive noise model (ANM)* (Peters et al. 2014), which is a popular and widely studied model in the literature.

To estimate a causal DAG from observational data, several algorithms have been proposed so far (Shimizu et al. 2011; Ghoshal and Honorio 2018; Zheng et al. 2018; Cai et al. 2018; Lachapelle et al. 2020; Zheng et al. 2020). In this paper, we focus on the *order-based approach* (Teyssier and Koller 2005), which first estimates a topological order of the underlying causal DAG and then prunes spurious edges from the fully-connected DAG induced by the estimated order. Existing studies often focus on the former step and proposed several methods for estimating a topological order, such as *CAM* (Bühlmann, Peters, and Ernest 2014), *RESIT* (Peters et al. 2014), *SCORE* (Rolland et al. 2022), and so on (Xu et al. 2024; Wang et al. 2021; Sanchez et al. 2023). For the latter step, most of these methods employ *CAM-pruning* (Bühlmann, Peters, and Ernest 2014) that identifies spurious edges by nonlinear feature selection based on GAMs. One exception is *DAS* proposed by (Montagna et al. 2023), which accelerates CAM-pruning by removing redundant candidate parents in advance by leveraging the score matching (Rolland et al. 2022). In contrast, this paper proposes an alternative pruning method that aims to address the limitations of CAM-pruning in terms of efficiency and accuracy. We demonstrated that our method can significantly accelerate existing order-based causal structure learning algorithms without compromising accuracy, enabling us to apply them to more high-dimensional settings.

Our SARTRE can be regarded as a new *nonlinear variable selection* method, as well as a new learning algorithm for *sparse additive models* (Ravikumar et al. 2009; Haris, Simon, and Shojaie 2022). It consists of the *randomized tree embedding* (Moosmann, Triggs, and Jurie 2006; Geurts, Ernst, and Wehenkel 2006; Feng and Zhou 2018) and *group lasso regression* (Yuan and Lin 2006; Massias, Gramfort, and Salmon 2018). While frameworks for learning GAMs with tree-based shape functions exist (Lou et al. 2013), they are not designed to encourage sparsity. Since variable selection plays a crucial role in many applications, our SARTRE has the potential to be applied to various tasks beyond causal structure learning (Marra and Wood 2011).

6 Conclusion

This paper proposed a new pruning method that enhances the efficiency and accuracy of nonlinear order-based causal structure learning. To address the limitations of the existing pruning method based on additive models with hypothesis testing, we introduced a new framework, named SARTRE, for learning a sparse additive model by combining the randomized tree embedding and group-wise sparse regression techniques. Our method can efficiently learn a sparse additive model for each variable and its candidate parents, which enables us to directly prune redundant edges without hypothesis testing. Experimental results demonstrated that our method was significantly faster than the existing methods while maintaining comparable or superior accuracy.

Limitations and Future Work One limitation of our method is that we need to determine some hyperparameters in advance. In our experiments, we used the same values for λ and l_j across all settings and confirmed that our method stably performed well in all situations. However, it would be beneficial to develop a method that can automatically tune these values based on the data. Another limitation is the lack of theoretical guarantees for our pruning quality. While we showed that our SARTRE has a rich representation ability in Proposition 1, guaranteeing the correctness of pruning remains an open problem. Finally, we need to examine our method in more general and practical settings. For example, investigating the performance of our method in the presence of *latent confounders* is important for future work.

Ethical Statement

Our proposed method, named SARTRE, is a new pruning algorithm for nonlinear causal structure learning from observational data. We believe that our method can be applied to enhance the understanding of complex systems across various fields, including biology, economics, and the social sciences. It enables researchers to discover causal relationships in data that may not be readily observable, leading to more informed decision-making. We acknowledge that the use of causal structure learning methods can have significant ethical implications, particularly in sensitive areas such as healthcare or criminal justice. To prevent potential misuse, we need to ensure that our method is used responsibly and ethically. Overall, we believe that our method can have a positive impact on society by enhancing the understanding of complex systems and facilitating better decision-making, provided it is used adequately.

Acknowledgments

We wish to thank Yuta Fujishige, Shun Yanashima, and Ryosuke Ozeki for making a number of valuable suggestions. We also thank the anonymous reviewers for their insightful comments.

References

Breiman, L. 2001. Random Forests. *Machine Learning*, 45(1): 5–32.

Bühlmann, P.; Peters, J.; and Ernest, J. 2014. CAM: Causal additive models, high-dimensional order search and penalized regression. *Annals of Statistics*, 42(6): 2526–2556.

Cai, R.; Qiao, J.; Zhang, Z.; and Hao, Z. 2018. SELF: Structural Equational Likelihood Framework for Causal Discovery. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 1787–1794.

Chickering, D. M. 1996. Learning Bayesian Networks is NP-Complete. In *Learning from Data: Artificial Intelligence and Statistics V*, 121–130.

Chickering, D. M. 2003. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3: 507–554.

Cybenko, G. 1989. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4): 303–314.

Feng, J.; and Zhou, Z.-H. 2018. AutoEncoder by Forest. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 2967–2973.

Friedman, J.; Hastie, T.; and Tibshirani, R. 2010. A note on the group lasso and a sparse group lasso. *arXiv*, arXiv:1001.0736.

Friedman, J. H. 2000. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics*, 29: 1189–1232.

Friedman, J. H.; and Popescu, B. E. 2008. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3): 916–954.

Geurts, P.; Ernst, D.; and Wehenkel, L. 2006. Extremely randomized trees. *Machine Learning*, 63(1): 3–42.

Ghoshal, A.; and Honorio, J. 2018. Learning linear structural equation models in polynomial time and sample complexity. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, 1466–1475.

Haris, A.; Simon, N.; and Shojaie, A. 2022. Generalized Sparse Additive Models. *Journal of Machine Learning Research*, 23(70): 1–56.

Hastie, T.; and Tibshirani, R. 1986. Generalized Additive Models. *Statistical Science*, 1(3): 297–310.

Hastie, T.; Tibshirani, R.; and Friedman, J. H. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer Series in Statistics. Springer.

Huang, B.; Zhang, K.; Lin, Y.; Schölkopf, B.; and Glymour, C. 2018. Generalized Score Functions for Causal Discovery. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1551–1560.

Lachapelle, S.; Brouillard, P.; Deleu, T.; and Lacoste-Julien, S. 2020. Gradient-Based Neural DAG Learning. In *Proceedings of the 8th International Conference on Learning Representations*.

Li, Y.; and Turner, R. E. 2018. Gradient Estimators for Implicit Models. In *Proceedings of the 6th International Conference on Learning Representations*.

Lou, Y.; Caruana, R.; Gehrke, J.; and Hooker, G. 2013. Accurate Intelligible Models with Pairwise Interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 623–631.

Marra, G.; and Wood, S. N. 2011. Practical variable selection for generalized additive models. *Computational Statistics & Data Analysis*, 55(7): 2372–2387.

Massias, M.; Gramfort, A.; and Salmon, J. 2018. Celer: a Fast Solver for the Lasso with Dual Extrapolation. In *Proceedings of the 35th International Conference on Machine Learning*, 3315–3324.

Montagna, F.; Noceti, N.; Rosasco, L.; Zhang, K.; and Locatello, F. 2023. Scalable Causal Discovery with Score Matching. In *Proceedings of the 2nd Conference on Causal Learning and Reasoning*, 752–771.

Moosmann, F.; Triggs, B.; and Jurie, F. 2006. Fast Discriminative Visual Codebooks using Randomized Clustering Forests. In *Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, 985–992.

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

Peters, J.; and Bühlmann, P. 2015. Structural Intervention Distance for Evaluating Causal Graphs. *Neural Computation*, 27(3): 771–799.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT press.

Peters, J.; Mooij, J. M.; Janzing, D.; and Schölkopf, B. 2014. Causal Discovery with Continuous Additive Noise Models. *Journal of Machine Learning Research*, 15(58): 2009–2053.

Ravikumar, P.; Lafferty, J.; Liu, H.; and Wasserman, L. 2009. Sparse Additive Models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 71(5): 1009–1030.

Rolland, P.; Cevher, V.; Kleindessner, M.; Russell, C.; Janzing, D.; Schölkopf, B.; and Locatello, F. 2022. Score Matching Enables Causal Discovery of Nonlinear Additive Noise Models. In *Proceedings of the 39th International Conference on Machine Learning*, 18741–18753.

Sachs, K.; Perez, O.; Pe'er, D.; Lauffenburger, D. A.; and Nolan, G. P. 2005. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721): 523–529.

Sanchez, P.; Liu, X.; O'Neil, A. Q.; and Tsaftaris, S. A. 2023. Diffusion Models for Causal Discovery via Topological Ordering. In *Proceedings of the 11th International Conference on Learning Representations*.

Shimizu, S.; Hoyer, P. O.; Hyvärinen, A.; and Kerminen, A. 2006. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *Journal of Machine Learning Research*, 7: 2003–2030.

Shimizu, S.; Inazumi, T.; Sogawa, Y.; Hyvärinen, A.; Kawahara, Y.; Washio, T.; Hoyer, P. O.; and Bollen, K. 2011. DirectLiNGAM: A Direct Method for Learning a Linear Non-Gaussian Structural Equation Model. *Journal of Machine Learning Research*, 12: 1225–1248.

Smith, S. M.; Miller, K. L.; Salimi-Khorshidi, G.; Webster, M.; Beckmann, C. F.; Nichols, T. E.; Ramsey, J. D.; and Woolrich, M. W. 2011. Network modelling methods for FMRI. *NeuroImage*, 54(2): 875–891.

Spirtes, P.; and Glymour, C. 1991. An algorithm for fast recovery of sparse causal graphs. *Social Science Computer Review*, 9: 67–72.

Teyssier, M.; and Koller, D. 2005. Ordering-based search: a simple and effective algorithm for learning Bayesian networks. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*, 584–590.

Wang, X.; Du, Y.; Zhu, S.; Ke, L.; Chen, Z.; Hao, J.; and Wang, J. 2021. Ordering-Based Causal Discovery with Reinforcement Learning. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, 3566–3573.

Xu, Z.; Li, Y.; Liu, C.; and Gui, N. 2024. Ordering-based causal discovery for linear and nonlinear relations. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems*, 4315–4340.

Yuan, M.; and Lin, Y. 2006. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1): 49–67.

Zheng, X.; Aragam, B.; Ravikumar, P.; and Xing, E. P. 2018. DAGs with NO TEARS: continuous optimization for structure learning. In *Proceedings of the 32nd Annual Conference on Neural Information Processing Systems*, 9492–9503.

Zheng, X.; Dan, C.; Aragam, B.; Ravikumar, P.; and Xing, E. 2020. Learning Sparse Nonparametric DAGs. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, 3414–3425.