Information-Theoretic Causal Discovery and Intervention Detection over Multiple Environments

Osman Mian[1], Michael Kamp[2], Jilles Vreeken[1]

[1] CISPA Helmholtz Center for Information Security.
[2] Institute for AI in medicine IKIM, Ruhr-University Bochum, and Monash University
osman.mian@cispa.de, michael.kamp@uk-essen.de, vreeken@cispa.de

Abstract

Given multiple datasets over a fixed set of random variables, each collected from a different environment, we are interested in discovering the shared underlying causal network and the local interventions per environment, without assuming prior knowledge on which datasets are observational or interventional, and without assuming the shape of the causal dependencies. We formalize this problem using the Algorithmic Model of Causation, instantiate a consistent score via the Minimum Description Length principle, and show under which conditions the network and interventions are identifiable. To efficiently discover causal networks and intervention targets in practice, we introduce the ORION algorithm, which through extensive experiments we show outperforms the state of the art in causal inference over multiple environments.

1 Introduction

We consider the setting where we have multiple datasets generated by a shared underlying causal mechanism, but where each dataset is collected over a different environment. That is, each dataset obtains observations over the same set of variables, but with a different source distribution, or, may be generated through an intervention upon the underlying mechanism. Our goal is to jointly discover the overall causal network as well as the local interventions without knowing which datasets are observational and which are interventional.

As a motivating example, suppose we are interested in learning the underlying causal process of some rare disease. A single hospital typically sees too few such patients as to collect sufficient data for drawing causal conclusions, and hence we will have to consider data collected at multiple hospitals. It is at best cumbersome to centralize the data due to privacy regulations. Even if we could centrally collect the data, by their location and specialization, every hospital will have a different distribution of patients, and because of difference in staff, machinery, etc., the parameters of the local data generating mechanisms will not all be exactly the same. If, for example, a certain test or drug is locally unavailable, the data collected there will be from an interventional distribution. Whether a dataset has been intervened upon is unknown in general and pool all data together in such cases can introduce bias in estimation (Lee and Tsui 1982; Tillman 2009).

While there exist approaches capable of discovering causal networks (Spirtes et al. 2000; Chickering 2002; Shimizu et al. 2006; Huang et al. 2018; Peters et al. 2014), they are designed to work only on a single dataset. Approaches that do take the multiple datasets into account work on strict assumptions such as having prior knowledge of intervention targets (Yang, Katcoff, and Uhler 2018; Hauser and Bühlmann 2012), can not match interventions on environments (Zhang et al. 2017) or impose strict assumptions on the underlying causal mechanisms that are unlikely to hold in practice (Shimizu 2012; Ghassami et al. 2017).

To discover causal networks using data over multiple environments, we build our approach on the algorithmic model of causality. We use the postulate of Algorithmic Markov Condition (AMC) (Janzing and Schölkopf 2010) stating that the true causal factorization of the joint distribution has the lowest Kolmogorov complexity, which allows us to uniquely identify a fully directed overall causal networks and local interventions. Kolmogorov complexity is not computable itself, but can be instantiated in a statistically well-founded manner using the Minimum Description Length (MDL) principle (Marx and Vreeken 2021).

We define a theoretically sound MDL score for jointly discovering the causal model and local interventions, and provide a practical greedy-algorithm to optimize our proposed score. We explicitly do not assume any prior knowledge of which datasets are observational or interventional and neither assume anything about the functional form of causal relationships between the variables. Our contributions are:

1. We build an approach to discover the overall causal network, the intervention targets within each environment, as well as the local causal networks for data collected over different environments.
2. We instantiate a consistent MDL-based score for nonlinear causal models and show under which assumptions it identifies fully directed causal networks and interventions.
3. To discover causal networks and interventions in practice, we propose an efficient greedy DAG search algorithm, called ORION. Through an extensive set of experiments we verify that it performs well in practice, outperforming the state of the art exact approaches in both causal discovery and identifying interventions over multiple environments.

2 Related Work

There exist many proposals for discovering causal networks from a single (typically observational) i.i.d. dataset (Spirtes et al. 2000; Chickering 2002; Huang et al. 2018; Compton et al. 2021), which discover partially directed causal networks. While Mian, Marx, and Vreeken (2021) propose an approach to discover fully directed networks, their method is restricted to a single dataset and can not handle interventions. Initial proposals that discover causal networks over multiple environments focused on single target variables (Peters, Bühlmann, and Meinshausen 2016; Yu et al. 2019) and can not trivially be extended to discover causal networks. Many methods assume we either know the intervention targets (Hauser and Bühlmann 2012; Triantafillou and Tsamardinos 2015; Yang, Katcoff, and Uhler 2018), or the environments that were intervened upon(Squires, Wang, and Uhler 2020; Brouillard et al. 2020). Recently, Faria, Martins, and Figueiredo (2022) proposed an approach to relax the assumption of known intervention environments. Approaches that do not need prior knowledge of interventions substitute it with other restrictive assumptions such as assuming a single type of intervention (Cooper and Yoo 1999; Kocaoglu et al. 2019) or fixing a functional form between cause and effect (Eaton and Murphy 2007; Shimizu 2012). In practice we often neither know which environments are interventional, nor do we know intervened variables, nor the causal functional forms.

The task of discovering causal networks over multiple environments without assuming any prior knowledge of interventions has been addressed by introducing an additional *context* variable that takes a fixed value within each environment (Zhang et al. 2017). While a single context variable allows to identify intervention targets across different environments, one can not single out the environment where the intervention happens. Mooij, Magliacane, and Claassen (2016) propose the unifying Joint Causal Inference (JCI) framework that can be implemented using any constraint-based causal discovery algorithm. JCI proposes to introduce one context variable per environment, thereby allowing localization of intervention targets within each context. JCI, however, outputs the overall global causal network and the intervention targets. It does not give us information about what are the local causal networks within environment, or what type of intervention has been performed. Finally, Jaber et al. (2020) recently provide a graphical characterization for testing whether two causal graphs with potentially different intervention targets belong to the same equivalence class. They, however, works under the assumption that the underlying structure stays the same for all the environments.

3 Preliminaries

Setup and Notation For a set of random variables, $\boldsymbol{X} = \{X_1, \ldots, X_m\}$ with $X_i \in \mathbb{R}$, a *Structural Causal Model* (SCM) (Pearl 2009) $\mathcal{S}$ models a joint distribution P over $\boldsymbol{X}$ corresponding to the observational distribution of the system. A causal DAG G over $\boldsymbol{X}$ is a graph in which the nodes represent random variables and edges identify the causal relationships as defined by $\mathcal{S}$. A directed edge between two variables $X_i \to X_j$ implies that X_i is a *direct cause* or *parent* of X_j. We denote the set of parents of X_j with pa_j and use $|\text{pa}_j|$ to denote the size of the parent-set. Given a sample $D \in \mathbb{R}^{m \times n}$ of size n from P, the goal of causal discovery is to identify the underlying causal *directed acyclic graph* (DAG) G entailed by $\mathcal{S}$ from this sample.

Under the assumptions of 1) causal faithfulness (Spirtes et al. 2000), 2) the causal Markov condition (Spirtes et al. 2000) and 3) causal sufficiency (Pearl 2009) it is possible to discover causal networks from observational data up to the Markov equivalence class (Glymour, Zhang, and Spirtes 2019). When we want to identify a fully oriented causal network we need additional assumptions (Peters, Janzing, and Schölkopf 2017), such as that the effect is a non-linear function of its causal parents with independent, additive Gaussian noise (Hoyer et al. 2009) or the assumption of low-noise between causal pairs (Marx and Vreeken 2019) which we elaborate in Sec. 4.

Under these assumptions, fully directed causal networks cannot only be identified, but also learned from data (Shimizu et al. 2006; Mian, Marx, and Vreeken 2021). Next, we show how the DAG G can be learned given a dataset D.

Information Theoretic Causal Discovery The main building block of the information theoretic model of causality is the algorithmic Markov condition (Janzing and Schölkopf 2010) which is based on Kolmogorov complexity. The Kolmogorov complexity of a finite binary string x is the *length* of the shortest binary program p^* for a universal Turing machine $\mathcal{U}$ that outputs x and *halts* (Kolmogorov 1965; Li and Vitányi 2009). This p^* is the length of the ultimate lossless compression of x. Similarly, the Kolmogorov complexity of a probability distribution P, $K(P)$, is the *length* of the shortest program that outputs $P(x)$ to precision q on input $\langle x, q \rangle$ (Li and Vitányi 2009). Formally stated,

$$K(P) = \min_{p \in \{0,1\}^*} \{|p| : |\mathcal{U}(p, x, q) - P(x)| \leq 1/q\} \ .$$

Using Kolmogorov complexity, Janzing and Schölkopf (2010) postulate the Algorithmic Markov Condition (AMC).

Postulate 1 ((Janzing and Schölkopf 2010)) *A causal DAG G over random variables $\boldsymbol{X}$ with joint density P is only acceptable if the shortest description of P factorizes as*

$$K(P(X_1, \ldots, X_m)) = \sum_{j=1}^{m} K(P(X_j \mid pa_j)) \ . \qquad (1)$$

which holds up to an additive constant.

Under this model the true DAG that generated D will have the lowest Kolmogorov complexity. Intuitively, this implies that the set pa_j most succinctly describes each $P(X_j|\cdot)$.

Due to, among others, the halting problem, Kolmogorov complexity is not computable. We can, however, approximate it from above through lossless compression (Li and Vitányi 2009). The Minimum Description Length (MDL) principle (Rissanen 1978; Grünwald 2007) provides a statistically well-founded framework to do so. Marx and Vreeken (2021) prove a formal connection between AMC and MDL by showing that the MDL formulation gives (on expectation) the same inference result as the original postulate. Therefore, in the limit $n \to \infty$, finding the true DAG can be achieved by finding the minimizer of a suitable lossless MDL score. Given a model class $\mathcal{M}$, MDL chooses the best model $M \in \mathcal{M}$ for data D as the one that minimizes, $L(D, M) = L(M) + L(D \mid M)$, where $L(M)$ is the length in bits of the description of M, and $L(D \mid M)$ is the length in bits of the description of data D given M.

To use MDL in practice we need to define a model class, and how to encode a model, resp. the data given a model, into bits. Our goal is to measure the *complexity* of a dataset under a model class after all. We are not concerned with the actual codes but rather only the optimal code *lengths* (Grünwald 2007). Hence, all logarithms are to base 2 and we use the common convention that $0 \log 0 = 0$.

Intervention Detection An intervention set Υ over an SCM $\mathcal{S}$ defines any external perturbation that inhibits the influence of one or more parents of any $X_i \in \boldsymbol{X}$, resulting in a new joint distribution $\widetilde{P}$ over $\boldsymbol{X}$. If we were to know the true causal DAG G^* that models the observational distribution over $\boldsymbol{X}$ and have infinite samples from some new environment $\widetilde{D}$, it is straightforward to discover if $\widetilde{D}$ was generated from the original DAG G^* or an intervened DAG $\widetilde{G}$: First, we would discover $\widetilde{G}$ over $\widetilde{D}$. We can then simply consider the difference between the edge-sets $\mathcal{E}(G^*) - \mathcal{E}(\widetilde{G})$ to discover what are the intervened variables, if any.

In practice, neither do we have infinite data, nor do we know G^* in advance. Even if we could learn G^* from limited data D, we first need to ensure that there are no interventions present in D. This results in a cyclic dependency as learning what interventions are present in the data was our goal in the first place. The key question we hence need to answer is: How can we, given only limited data from multiple environments, *simultaneously* discover the true overall causal network, the local causal structures as well as the intervention targets within each environment? This we discuss next.

4 Causal Discovery from Data Drawn from Multiple Environments

In this section we build on the algorithmic Markov condition described in Sec. 3 to identify the global resp. local causal models, as well as the intervention targets. Formally, our problem statement is:

Problem Statement 1 *Given samples $\boldsymbol{D} = \{D^1, \ldots, D^d\}$ over d environments that share a common SCM. Our goal is to (a) identify a single causal DAG G^* representing the true SCM; (b) identify which $D^k \in \boldsymbol{D}$ are interventional and which $X_i \in D^k$ are intervened upon; and (c) identify the local causal network for each D^k.*

To address this, we first define our causal model, list down the assumptions necessary to prove identifiability and present a novel score. Then we show that the optimizer of this score identifies the true causal model and interventions in the limit.

4.1 Causal Model and Assumptions

We consider a setup where in each environment k, the value of each variable X_i is determined by a *non-linear* function f_i^k over its causal parents and additive independent Gaussian noise term with zero mean and unit variance N_i, regulated by a scaling factor α_i^k. For X_i in environment k we have

$$X_i := f_i^k(\mathrm{pa}_i) + \alpha_i^k \cdot N_i \; . \tag{2}$$

We assume that all N_i are jointly independent and that $N_i \perp\!\!\!\perp \mathrm{pa}_i$ for all $X_i \in D^k$. We assume that the number of parameters required to non-parametrically model f_i^k are upper-bounded by $O(\log n)$ (Mian, Marx, and Vreeken 2021).

Assumptions for Identifying Markov Equivalence Classes To discover causal networks up to Markov equivalence class we need to assume 1) the causal Markov condition, 2) the causal faithfulness (Spirtes et al. 2000), and 3) causal sufficiency (Pearl 2009). These assumptions allows us to guarantee identifiability up to the Markov equivalence class of DAGs, and not just partial ancestral graphs (PAGs) (Spirtes, Meek, and Richardson 1999).

Assumptions for Identifying Fully Oriented Networks To ensure that we can orient edges between any pair of variables, and not just the edges coming into colliders, as is the case with the Markov equivalence class, we additionally need the low-noise assumption, meaning that the noise variance is sufficiently small for the causal *pairs* within a Markov equivalence class (Blöbaum et al. 2018) i.e. $\boldsymbol{\alpha} \rightarrow \mathbf{0}$, where $\boldsymbol{\alpha}$ is the vector consisting of scaling factors α_i^k for the bivariate causal edges and $\mathbf{0}$ is the null vector. Alternatively, we can make the assumption that these bivariate causal relationships are non-invertible. In this work, we make the low-noise assumption because it also covers the class of non-invertible causal relationships and is therefore a more general case of the two. This, however, does not imply that the causal relationships are deterministic. For an extensive discussion on the low-noise assumption see (Blöbaum et al. 2018)[Sec. 3].

Assumptions for Identifying Interventions We assume that the true underlying causal network G that generates the data remains the same for all environments unless it is specifically changed by either (i) Hard-Interventions $\mathrm{HI}(X_j)$; or (ii) inhibiting Soft-Interventions $\mathrm{SI}(X_j)$. A hard intervention on variable X_j eliminates the effect of pa_j on X_j, whereas a soft-intervention causes a *mechanism change* that sets the effect of a subset of pa_j to 0.

4.2 Encoding the Causal Model

To instantiate AMC (Eq. (1)) for our causal model (Eq. (2)) we need to define a lossless MDL score (Marx and Vreeken 2021). The model class $\mathcal{M}$ that we consider for our proposed MDL score consists of all possible DAGs over $\boldsymbol{X}$, the set of local DAGs each environment, as well as the SCM that models f_i^k for all X_i in each $D^k \in \boldsymbol{D}$. The correct model $M \in \mathcal{M}$ is therefore one that minimizes $L(\boldsymbol{D}, M)$ such that

$$\begin{aligned} M^* &= \underset{M \in \mathcal{M}}{\operatorname{argmin}}\, L(\boldsymbol{D}, M) \\ &= \underset{M \in \mathcal{M}}{\operatorname{argmin}} \left(L(M) + \sum_{k=1}^{d} \sum_{i=1}^{m} L(X_i^k | pa_i^k, f_i^k) \right) \\ &= \underset{M \in \mathcal{M}}{\operatorname{argmin}} \left(L(M) + \sum_{k=1}^{d} \sum_{i=1}^{m} L(\epsilon_{i,k}) \right) \end{aligned}$$

where pa_i^k are parents of variable X_i in dataset k according to the model M. We reformulate $L(X_i^k|pa_i^k, f_i^k)$ in the above equation by $L(\epsilon_i^k)$ to highlight that encoding each X_i once f_i^k and the parents are specified, comes down to storing the residuals $\epsilon_{i,k}$. We define the cost of the model as

$$L(M) = L_{str}(M) + \sum_{k=1}^{d} L_{mec}(M^k|M) \ ,$$

where L_{str} is the cost of storing the network structures and L_{mec} is the cost of storing the SCM once the structure is specified. Next, we describe what each of these costs are.

Structure The structure cost consists of the number of bits required to encode the global causal network as well as the interventions present in each environment. Formally we have

$$L_{str}(M) = L(G^*) + \sum_{k=1}^{d} L(G^k|G^*) \ ,$$

where we first encode the global causal network G^*, and for each G^k what are the interventions on G^*. Formally stated

$$L(G^*) = L_{\mathbb{N}}(d)+L_{\mathbb{N}}(m)+\sum_{i=1}^{m} L_{\mathbb{N}}(|pa_i|)+\log \binom{m}{|pa_i|} \ ,$$

where we first encode the number of environments, resp. variables, using $L_{\mathbb{N}}$, the optimal encoding for integers $z \geq 0$ (Rissanen 1983). It is defined as $L_{\mathbb{N}}(z) = \log^* z + \log c_0$, where $\log^* z = \log z + \log\log z + \ldots$ and we consider only the positive terms, c_0 is a normalization constant to ensure the Krafft-inequality holds (Krafft 1949). Then, for each of the m variables, we encode the number of parents $|pa_i|$ and identify pa_i from m using $\log \binom{m}{|pa_i|}$ bits.

Next we encode the local networks G^k once the interventions over G^* are provided, i.e. $L(G^k|G^*)$ is defined as

$$L(G^k|G^*) = \log(m) + \log \binom{m}{\tilde{m}^k}$$
$$+ \sum_{X_i \in \widetilde{X}^k} \log(|pa_i|) + \log \binom{|pa_i|}{|pa_i^k|} \ .$$

For each local network, we encode the number, $\tilde{m}$ and identity $\widetilde{X}^k$ of intervened variables. Then, for each intervened variable, we identify the its active set of parents.

Combining the above, we have a lossless code for the causal structure.

Mechanisms Next we define how to encode an SCM over M. Effectively we have to encode the function f_i^k for all X_i in each $D^k \in \boldsymbol{D}$. This is defined as

$$L_{mec}(M^k|M) = \sum_{i=1}^{m} L(f_i^k) \ .$$

Our causal model makes no assumption on the functional form of the causal relationship. We model each f_i^k *non-parametrically*. In particular we use multivariate regression splines (Friedman 1991) of the form $X_i := \sum_{j=1}^{|H|} f_j(\mathcal{P}_j)$, where f_j is a hinge function applied to a subset of X_i's parents $\mathcal{P}_j$ with size $|\mathcal{P}_j|$. A hinge function is of the form $f(\mathcal{P}) = a \cdot \prod_{t=1}^{T} \max(0, g_t(\mathrm{pa}_t) - b_t)$, where T denotes the number of multiplicative terms in the hinge, $\mathrm{pa}_t \in \mathcal{P}$ is the parent associated with the t-th term, and g_t is a non-linear transformation from a finite function class $\mathcal{F}$ applied to pa_t. The cost to store the causal mechanism using multivariate regression splines can then be defined as

$$L(f) = \ L_{\mathbb{N}}(|H|) + \sum_{h_j \in H} \Big[\ L_{\mathbb{N}}(T_j) + \log \binom{|\mathcal{P}| + T_j - 1}{T_j}$$
$$+ T_j \log(|\mathcal{F}|) + L_p(\theta_j) \ \Big] \ .$$

We use $L_{\mathbb{N}}$ to encode the number of hinges. Then for each hinge, we encode the number of terms per hinge, the correct assignment of terms T_j to parents in $\mathcal{P}$, the number of bits to identify non-linear transformations used for each term in the hinge, and parameters θ_j associated with th j-th term. We encode the parameters θ_j using $L_p(\theta_j)$ (Marx and Vreeken 2017) formally defined as

$$L_p(\theta) = \sum_{i=1}^{|\theta|} 1 + L_{\mathbb{N}}(z_i) + L_{\mathbb{N}}(\lceil \theta_i \cdot 10^{z_i} \rceil) \ ,$$

where z_i is the smallest integer such that $|\theta_i| \cdot 10^{z_i} \geq 10^p$. Simply put, $p = 2$ implies that we consider the first two digits of the parameter. For each parameter we encode the sign using 1 bit, encode the shift z_i and the shifted parameter θ_i. We work with fixed precision for parameters θ_i, meaning that L_p is computed in constant time w.r.t sample size.

Residuals As a final step to obtaining a lossless score, we need to encode the noise that remains in the system once the specified model has captured the structure and generating mechanism of the data. Since we use regression functions, we aim to minimize the variance of the residual, and hence encode the residual ϵ as Gaussian distributed with zero-mean (Grünwald 2007), that is

$$L(\epsilon_{i,k}) = \frac{n}{2} \left(\frac{1}{\ln 2} + \log 2\pi\hat{\sigma}_{i,k}^2 \right) \ ,$$

where we compute the empirical variance $\hat{\sigma}_{i,k}^2$ from the residual, $\epsilon_{i,k}$.

Combining all of the above, we have a lossless MDL score by which we can instantiate the AMC. Next we establish theoretical guarantees entailed by the defined causal model and prove that the minimizer of $L(\boldsymbol{D}, M)$ identifies the correct causal network and interventions in the limit.

4.3 Asymptotic Guarantees

In the following we show that the proposed score is consistent when $n \to \infty$. We show that under the assumptions described in Sec. 4.1, it identifies hard interventions as well as inhibiting soft-interventions. We provide all the proofs in Appendix A.

We begin by showing that missing edges in local causal networks are the result of interventions.

Lemma 1 *$\forall i, k$ $\textsc{Hi}(X_i^k) \iff pa_i^k = \emptyset$, and $\textsc{Si}(X_i^k) \iff pa_i^k \subset pa_i$*

To provide further identifiability results we first state the definition of a *conservative* set of interventions as stated by Hauser and Bühlmann(Hauser and Bühlmann 2012).

Definition 2 ((Hauser and Bühlmann 2012)) *A set of interventions $\boldsymbol{\Upsilon}$ is conservative, if $\forall X_i \in \bigcup_{k=1}^{d} \Upsilon^k, \exists \Upsilon^k \in \Upsilon$ such that $X_i \notin \Upsilon^k$.*

Simply put, a set of interventions $\boldsymbol{\Upsilon}$ is conservative if for each variable X_i we can find at least one environment in which it is not intervened upon ($X_i \notin \Upsilon^k$). Let G^* be the true global network and G^k be the network discovered for environment k.

Lemma 3 *If $\boldsymbol{\Upsilon}$ is conservative, $\bigcup_{k=1}^{d} G^k = G^*$, if $\boldsymbol{\Upsilon}$ is non-conservative, $\bigcup_{k=1}^{d} G^k \subseteq G^*$.*

Next, we provide our main result. We can show the following best resp. worst case result that we can guarantee for the causal model defined in Eq. (2).

Theorem 4 *Let $\mathcal{Y}$ be the set of all effects such that $\forall Y_i \in \mathcal{Y}$, $|pa_i| = 1$. If $\forall Y_i, k\ \alpha_i^k \to 0$, $L(\boldsymbol{D}, M)$ will be the lowest for the true fully-oriented causal network.*

Moreover we can still identify the correct Markov equivalence class, even when the low noise assumption is violated,

Theorem 5 *$L(\boldsymbol{D}, M)$ correctly identifies the collider structures in the underlying causal network.*

Proof sketch For an intuitive explanation of our main result, consider Thm. 5 first. Identifying collider structure means that our score identifies causal DAGs up to Markov equivalence class at the very least. This implies that any undirected edges that exist in the final network are between pairs of variables that are not colliders. For such case, our causal model simplifies to the pair-wise model of Marx and Vreeken (Marx and Vreeken 2019). They prove that under the low-noise assumption, orientation of such pair-wise edges is identifiable using an L_0 regularized score (e.g. BIC). Meaning, for the pair-wise model between variables X and Y, the BIC score for regressing Y onto X, resp. X onto Y, will be highest in the causal direction. Next, note that the BIC score is equal to the negative of the MDL criterion. Thus, if we were to score *all* Markov equivalent DAGs using an MDL based L_0 regularized score, the causal one will obtain the lowest score. Consequently, to prove Thm. 5, we reformulate $L(\boldsymbol{D}, M)$ to show that it is a valid L_0 regularized score. Using this score in conjunction with the low-noise assumption stated in Sec 4.2 lets us identify any remaining bivariate cases in the causal network, which proves Thm. 4.

It is worth noting that our proposed score identifies the fully oriented causal network, it neither requires using distribution-shifts nor introducing additional context variables to orient any remaining edges. These theoretical guarantees, however, only hold if we score all possible DAGs over the data. This quickly becomes infeasible for large graphs. Indeed, finding the exact Bayesian network is known to be NP-hard (Chickering, Heckerman, and Meek 2004). Hence, we propose a practical approach to minimizing $L(\boldsymbol{D}, M)$.

Algorithm 1: The ORION Algorithm

```
Input: Datasets $\boldsymbol{D}$ over $\boldsymbol{X}$
Output: Array of causal networks $\boldsymbol{G}$
for $k = 1 \dots d$ do
    $G^k \leftarrow \emptyset$
$\boldsymbol{G} \leftarrow [G^1, \dots G^d]$
repeat
    $\boldsymbol{G} \leftarrow$ FORWARDSEARCH($\boldsymbol{G}, \boldsymbol{D}$)
    $\boldsymbol{G} \leftarrow$ BACKWARDSEARCH($\boldsymbol{G}, \boldsymbol{D}$)
until convergence;
return $\boldsymbol{G}$
```

5 Practical Algorithm

In this section we present a practical algorithm ORION for discovering causal DAGs from multivariate continuous valued data over multiple environments. ORION greedily adds and removes edges to the global resp. local causal networks such that it reduces $L(\boldsymbol{D}, M)$ most. Similar to GES, it performs forward and backward search, repeated until convergence. We provide the algorithm outline in Alg. 1 and give detailed pseudocode in Appendix C. It learns a causal network by iteratively adding and removing edges to the global structure, and encoding interventions for the datasets that reject the globally introduced edges. As output, it returns the (intervened) local causal networks. We take union over these networks to reconstruct the predicted global causal network (Lem. 3) and take the difference between the edge-sets of global and local causal networks to determine the intervention targets (Lem. 1). As our score is lower-bounded at 0, and we only take steps that reduce our score, it is guaranteed to converge. Even though the guarantees of greedy DAG search are limited to causal trees, we show in Sec. 6 that ORION outperforms state-of-the-art exact search algorithms. Next, we describe the ranking mechanism and the search phases.

Edge gain To calculate the gain provided by each edge, we first measure the bits that we save by adding an edge in the current model. Formally, let $e_{ij} = X_i \rightarrow X_j$, and M be the current model. We write $M \oplus e_{ij}$ to denote the model with edge e_{ij} included. We define the absolute gain in bits δ associated with edge e_{ij} as

$$\delta(e_{ij}) = \max\{0, L(\boldsymbol{D}, M) - L(\boldsymbol{D}, M \oplus e_{ij})\} \ .$$

Next, we calculate the true gain for this edge by calculating the relative bits we gain over adding this edge in the opposite direction. Formally,

$$\psi(e_{ij}) = \delta(e_{ij}) - \delta(e_{ji}) \ .$$

Intuitively, the higher the value of $\psi(e_{ij})$, the more certain we are that we inferred the correct direction for this edge. This is motivated by the no-hypercompression inequality (Grünwald 2007), which we use to test the significance of each edge. Let $s = \psi(e)$, the probability of gaining s bits over the null model is less than or equal to 2^{-s}. If we find that the gain for an edge is not significant— i.e. 2^{-s} is greater than the desired significance threshold— we do not add this edge.

Table 1: [Lower is Better] Averaged normalized Sid for synthetic data with $m = 10$. Intervals indicates the best, resp. worst possible intervention distance for methods that output the Markov equivalence class of the causal network.

d	Orion	Lingam	Jci-Pc	CdNod
3	**0.45**	0.58	[0.47, 0.67]	[0.48, 0.55]
5	**0.44**	0.55	[0.45, 0.67]	[0.44, 0.48]
7	**0.42**	0.53	[0.42, 0.65]	[0.56, 0.66]
9	**0.43**	0.52	[0.44, 0.63]	[0.60, 0.70]

Forward Search In forward search, we maintain a priority queue containing the edges e_{ij} ordered by the gain in bits $\psi(e_{ij})$, when adding the edge to the model. We iteratively build the causal graph by adding the highest ranked edge from the priority queue to the global causal DAG. We reject edges that introduce cycles in the network. Once an edge e_{ij} is added to the network, we re-rank all the candidate edges associated with variables X_j in the priority queue. We repeat this until all the edges have been evaluated and no edge addition provides gain anymore.

We introduce each edge as part of the global network which means that the structure cost is shared across datasets. Each of the datasets, therefore, only need to pay a discounted cost of storing their causal mechanism in order to include this edge. If the discounted cost is not enough to register a gain, an intervention is encoded for this dataset.

Backward Search Since we greedily add edges during the forward search phase, some parents of variable X_j may become redundant as forward search progresses. This is because a subset of these parents may be able to explain X_j better. To remove these redundant parents, we need a backward search. We iteratively remove that edge from the network which improves score the most. We remove edges until no edge removal improves $L(\boldsymbol{D}, M)$ anymore.

Complexity Analysis We first make a pass over the entire edge-set for each environment to determine the initial edge gains. This requires $\mathcal{O}(cdm^2 \log m)$ steps where c denotes the complexity of the regression approach that is used. In forward search, each edge can lead to at most $m - 1$ ranking updates, each of which require $\mathcal{O}(\log m)$ time when priority queue is implemented as a heap. Resulting in a complexity of $\mathcal{O}(cdm^3 \log m)$. The backwards search has a similar upper bound of $\mathcal{O}(cdm^3 \log m)$. Hence, the overall complexity is in $\mathcal{O}(cdm^3 \log m)$. Orion compares favorably to the worst-case complexities of Pc , $\mathcal{O}(2^m)$, Ges, $\mathcal{O}(2^m)$, CdNod, $\mathcal{O}(n^3)$. Orion is inherently parallelizable over both edges and environments, therefore quite fast in practice.

6 Evaluation

In this section we empirically evaluate Orion, we are mainly interested in answering the following three questions – (1) Does Orion accurately discover causal networks over data from multiple environments? (2) How well does Orion perform on real world networks where our assumptions may not hold? and (3) Does Orion reliably identify intervention targets? We first describe our experimental setup and then answer these questions in the subsequent set of experiments.

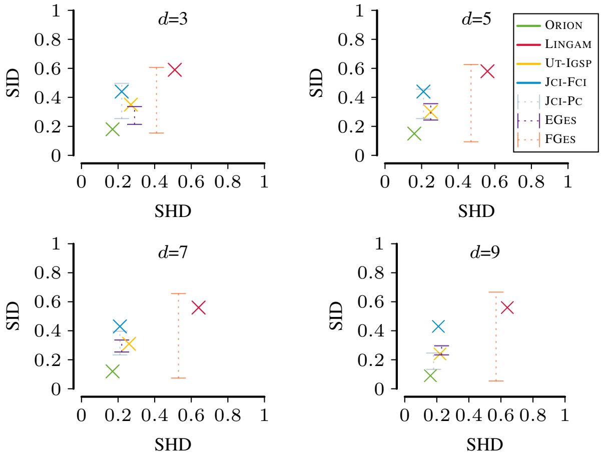


Figure 1: [Closer to origin is better] Comparison of normalized Shd and Sid when all environments contain data from a different intervention distribution over the same causal network. Dotted lines indicate the uncertainty interval over Sid for Jci-Pc, EGes and FGes.

Setup We compare to state-of-the-art approaches from the classes of ANM, constraint, and score-based methods. As the representative ANM-based method, we compare to Direct-Lingam (Shimizu 2012) which is an extension of the original Lingam (Shimizu et al. 2006) to multiple datasets. For constraint-based methods, we compare to CdNod (Zhang et al. 2017), and to the Jci framework of (Mooij, Magliacane, and Claassen 2016) using Pc (Spirtes et al. 2000) resp. Fci (Spirtes, Meek, and Richardson 1999). For score-based approaches, we compare to the permutation-based greedy search approach, Ut-Igsp (Squires, Wang, and Uhler 2020), the Ges algorithm (Chickering 2002; Ramsey et al. 2017) using the two-layer approach proposed by Eaton and Murphy (2007), which we refer to as EGes. As baseline, we compute results over vanilla fast-Ges (FGes) (Ramsey et al. 2017) by taking a union over locally discovered networks.

We evaluate the quality of the discovered networks in terms of structural similarity using the Structural Hamming Distance (Shd) (Kalisch and Bühlmann 2007) which measures the number of edges in which two networks differ. Shd, however, tells us nothing about the difference in networks' causal implications. To measure this causal similarity, we use the Structural Intervention Distance (Sid) (Peters and Bühlmann 2015). Sid counts those pairs of variables X_i and X_j, such that the effect experienced by X_j due to an intervention on X_i differs between two networks. For comparability over different datasets, we normalize Shd and Sid between 0 and 1, we give the unnormalized scores in Appx. D. To avoid practical issues like var-sortability (Reisach, Seiler, and Weichwald 2021), we standardize all data. We provide the full experimental setup in Appx D and make our code and data available in supplementary material.

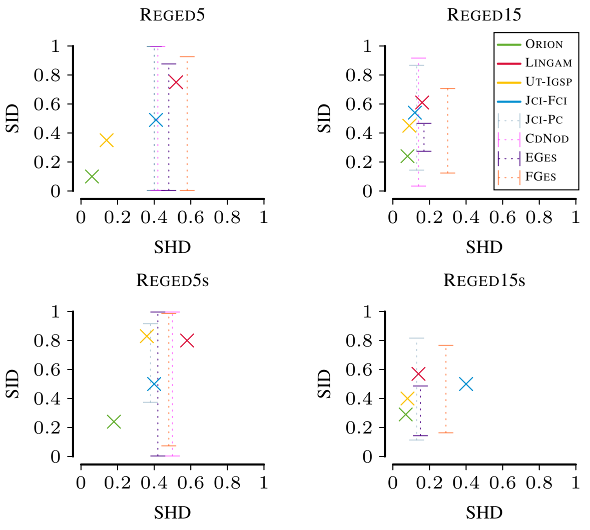


Figure 2: [Closer to origin is better] Comparison of normalized Shd and Sid for the Reged networks without selection bias (Reged5, Reged15) and with selection bias (Reged5s, Reged15s). Dotted lines indicate the interval over Sid for Jci-Pc, EGes and CdNod.

Q1. Does Orion accurately discover causal networks over data from multiple environments? We start with a simple setting where we generate multiple datasets using the same underlying distribution. We simulate DAGs using the Erdős-Rényi model. We model effect as a function of its causes using polynomial functions in half of the cases. For other half we use randomly initialized 2-layer neural networks to model the mechanism. We average the resulting Sid over 100 different runs and report the results in Table. 1. We omit Jci-Fci because it almost always returns empty networks, and FGes it reports Sid intervals too wide to convey meaningful information. We find that Orion reports the best Sid, at least as good as the lowest score over the equivalence classes that Jci-Pc resp. CdNod report.

Next, and more interestingly, we generate each environment using different intervention distributions from a fixed underlying causal network. This means that the data for each environment comes from a different (sub)network, about which we know neither the type nor the targets of intervention. We report the results in Fig. 1 where we see that Orion performs best. CdNod is unable to handle the cases involving hard interventions.

Q2. How well does Orion perform when assumptions may not hold? To this end, we use the re-simulated Lung-cancer gene expression, Reged network (Statnikov et al. 2015). We extract two non-overlapping connected components of 5 resp. 15 variables, which we refer to as Reged5 and Reged15. For both networks, we randomly divide the data into 3 environments containing 250 samples each.

Next, we introduce selection bias in the data by sorting on one of the variable and dividing the resulting dataset into three partially overlapping datasets of 200 samples each. We repeat this for each variable thereby giving us a total of 5 resp. 15 separate experiment instances for each network. We refer to these datasets as Reged5s resp. Reged15s.

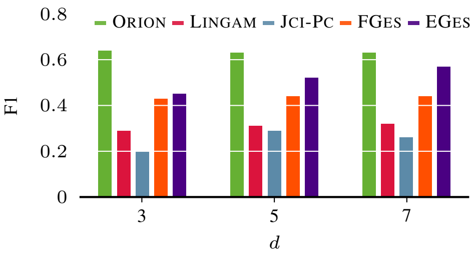


Figure 3: [Higher is better] F1 scores for Orion, Lingam, Jci-Pc, EGes and FGes for identifying intervention targets in synthetic data over different environment sizes, d. We omit CdNod as it does not contain a mechanism to identify intervention targets within each environment.

We show the results for both aforementioned setups in Fig. 2 where we see that Orion performs the best overall. Moreover, we see that EGes, CdNod and Jci-Pc have very wide Sid intervals, which restricts us from drawing useful causal conclusions from the discovered networks.

Q3. Can Orion reliably identify intervention targets? We test how well Orion can identify both direct and indirect intervention targets over multiple environments. We use the same structure as used by Zhang et al. (2017) for their experiments and report the F1-scores for this experiment in Fig. 3. We see that Orion gets an F1-score average of 0.63, which is twice as good as Lingam and Jci-Pc. Surprisingly, FGes, although only a baseline, performs better than both Lingam and Jci-Pc.

7 Discussion and Conclusion

We proposed novel scores for the discovery of causal networks over multiple environments based on the algorithmic Markov condition and its approximation via MDL. Our analysis proved that optimizing this score identifies the true DAG and all local interventions in the limit. This allows us to simultaneously discover the underlying causal mechanism and local interventions over multiple datasets. We proposed a practical algorithm Orion which, through extensive experiments, we showed that it outperforms the state of the art at discovering the true causal networks given multiple datasets, even when all the environments contain data generated from unknown intervention distributions over the same network, and reliably identifies intervention targets.

Although non-trivial, it is a promising direction to investigate implementing the GES (Chickering 2002) procedure using our score as a line of future work. Such an implementation will extend all our theoretical guarantees to the proposed implementation, at the expense of the worst-case runtime becoming exponential in the number of variables. Currently we are investigating evolving our proposed score to handle edge-introducing interventions alongside inhibiting interventions. Maintaining identifiabilty guarantees while doing so is a challenging yet worthwhile line of future work.

References

Blöbaum, P.; Janzing, D.; Washio, T.; Shimizu, S.; and Schölkopf, B. 2018. Cause-effect inference by comparing regression errors. In *International Conference on Artificial Intelligence and Statistics*, 900–909. PMLR.

Brouillard, P.; Lachapelle, S.; Lacoste, A.; Lacoste-Julien, S.; and Drouin, A. 2020. Differentiable causal discovery from interventional data. *Advances in Neural Information Processing Systems*.

Chickering, D. M. 2002. Optimal structure identification with greedy search. *JMLR*, 3: 507–554.

Chickering, M.; Heckerman, D.; and Meek, C. 2004. Large-sample learning of Bayesian networks is NP-hard. *JMLR*, 5.

Compton, S.; Kocaoglu, M.; Greenewald, K.; and Katz, D. 2021. Entropic causal inference: Identifiability and finite sample results. *arXiv preprint arXiv:2101.03501*.

Cooper, G. F.; and Yoo, C. 1999. Causal discovery from a mixture of experimental and observational data. *UAI*.

Eaton, D.; and Murphy, K. 2007. Exact Bayesian structure learning from uncertain interventions. In *AISTATS*, 107–114. PMLR.

Faria, G. R. A.; Martins, A.; and Figueiredo, M. A. 2022. Differentiable causal discovery under latent interventions. In *Conference on Causal Learning and Reasoning*, 253–274. PMLR.

Friedman, J. H. 1991. Multivariate adaptive regression splines. *The annals of statistics*, 1–67.

Ghassami, A.; Salehkaleybar, S.; Kiyavash, N.; and Zhang, K. 2017. Learning causal structures using regression invariance. *arXiv preprint arXiv:1705.09644*.

Glymour, C.; Zhang, K.; and Spirtes, P. 2019. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*.

Grünwald, P. 2007. *The Minimum Description Length Principle*. MIT Press.

Hauser, A.; and Bühlmann, P. 2012. Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs. *JMLR*, 13(1): 2409–2464.

Hoyer, P. O.; Janzing, D.; Mooij, J. M.; Peters, J.; and Schölkopf, B. 2009. Nonlinear causal discovery with additive noise models. In *NIPS*, 689–696.

Huang, B.; Zhang, K.; Lin, Y.; Schölkopf, B.; and Glymour, C. 2018. Generalized Score Functions for Causal Discovery. In *KDD*. ACM.

Jaber, A.; Kocaoglu, M.; Shanmugam, K.; and Bareinboim, E. 2020. Causal discovery from soft interventions with unknown targets: Characterization and learning. *Advances in neural information processing systems*, 33.

Janzing, D.; and Schölkopf, B. 2010. Causal Inference Using the Algorithmic Markov Condition. *IEEE TIT*, 56(10): 5168–5194.

Kalainathan, D.; and Goudet, O. 2019. Causal Discovery Toolbox: Uncover causal relationships in Python. *arXiv preprint arXiv:1903.02278*.

Kalisch, M.; and Bühlmann, P. 2007. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *JMLR*, 8(Mar): 613–636.

Kocaoglu, M.; Shanmugam, K.; Jaber, A.; and Bareinboim, E. 2019. Characterization and learning of causal graphs with latent variables from soft interventions. *Advances in neural information processing systems*.

Kolmogorov, A. 1965. Three Approaches to the Quantitative Definition of Information. *Problemy Peredachi Informatsii*, 1(1): 3–11.

Krafft, L. G. 1949. *A device for quantizing, grouping, and coding amplitude-modulated pulses*. Ph.D. thesis, Massachusetts Institute of Technology.

Lee, S.-Y.; and Tsui, K.-L. 1982. Covariance structure analysis in several populations. *Psychometrika*, 47.

Li, M.; and Vitányi, P. 2009. *An Introduction to Kolmogorov Complexity and its Applications*. Springer.

Marx, A.; and Vreeken, J. 2017. Telling Cause from Effect using MDL-based Local and Global Regression. In *ICDM*, 307–316. IEEE.

Marx, A.; and Vreeken, J. 2019. Identifiability of Cause and Effect using Regularized Regression. In *KDD*. ACM.

Marx, A.; and Vreeken, J. 2021. Formally Justifying MDL-based Inference of Cause and Effect. *arXiv preprint arXiv:2105.01902*.

Mian, O.; Marx, A.; and Vreeken, J. 2021. Discovering fully oriented causal networks. In *AAAI*.

Mooij, J. M.; Magliacane, S.; and Claassen, T. 2016. Joint causal inference from multiple contexts. *JMLR*, 21.

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.; Bühlmann, P.; and Meinshausen, N. 2016. Causal inference by using invariant prediction: identification and confidence intervals. *J. R. Statist. Soc. B*, 947–1012.

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

Peters, J.; Mooij, J. M.; Janzing, D.; and Schölkopf, B. 2014. Causal Discovery with Continuous Additive Noise Models. *JMLR*, 15.

Ramsey, J.; Glymour, M.; Sanchez-Romero, R.; and Glymour, C. 2017. A million variables and more: the Fast Greedy Equivalence Search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *J. Data Sci. Anal.*

Reisach, A.; Seiler, C.; and Weichwald, S. 2021. Beware of the Simulated DAG! Causal Discovery Benchmarks May Be Easy to Game. *Advances in Neural Information Processing Systems*, 34.

Rissanen, J. 1978. Modeling by shortest data description. *Automatica*, 14(1): 465–471.

Rissanen, J. 1983. A Universal Prior for Integers and Estimation by Minimum Description Length. *Annals Stat.*, 11(2): 416–431.

Shimizu, S. 2012. Joint estimation of linear non-Gaussian acyclic models. *Neurocomputing*, 81.

Shimizu, S.; Hoyer, P. O.; Hyvärinen, A.; and Kerminen, A. 2006. A Linear Non-Gaussian Acyclic Model for Causal Discovery. *JMLR*, 7.

Spirtes, P.; Glymour, C. N.; Scheines, R.; and Heckerman, D. 2000. *Causation, prediction, and search*. MIT Press.

Spirtes, P.; Meek, C.; and Richardson, T. 1999. An algorithm for causal inference in the presence of latent variables and selection bias. *Computation, causation, and discovery*, 21.

Squires, C.; Wang, Y.; and Uhler, C. 2020. Permutation-based causal structure learning with unknown intervention targets. In *Conference on Uncertainty in Artificial Intelligence*, 1039–1048. PMLR.

Statnikov, A.; Ma, S.; Henaff, M.; Lytkin, N.; Efstathiadis, E.; Peskin, E. R.; and Aliferis, C. F. 2015. Ultra-Scalable and Efficient Methods for Hybrid Observational and Experimental Local Causal Pathway Discovery. *JMLR*, 16: 3219–3267.

Strobl, E. V.; Zhang, K.; and Visweswaran, S. 2019. Approximate kernel-based conditional independence tests for fast non-parametric causal discovery. *Causal Inf.*, 7(1).

Tillman, R. E. 2009. Structure learning with independent non-identically distributed data. In *ICML*, 1041–1048.

Triantafillou, S.; and Tsamardinos, I. 2015. Constraint-based causal discovery from multiple interventions over overlapping variable sets. *JMLR*, 16(1): 2147–2205.

Yang, K.; Katcoff, A.; and Uhler, C. 2018. Characterizing and learning equivalence classes of causal dags under interventions. In *ICML*, 5541–5550. PMLR.

Yu, K.; Liu, L.; Li, J.; Ding, W.; and Le, T. D. 2019. Multisource causal feature selection. *IEEE TPAMI*.

Zhang, K.; Huang, B.; Zhang, J.; Glymour, C.; and Schölkopf, B. 2017. Causal discovery from nonstationary/heterogeneous data: Skeleton estimation and orientation determination. In *IJCAI*.

A Proofs

Lemma 1 $\forall i, k\ \ \textsc{Hi}(X_i^k) \iff pa_i^k = \emptyset$ *and* $\textsc{Si}(X_i^k) \iff pa_i^k \subset pa_i$

Proof: Assume that we are given the true causal network G^* for an SCM as well as the dataset D^k over the same SCM for which $\textsc{Si}(X_i)$ holds.

First, we prove the direction $\mathrm{pa}_i^k \subset \mathrm{pa}_i^* \longrightarrow \textsc{Si}(X_i^k)$ for Lemma 2. Assume that $pa_i^k \subset pa_i^*$ holds but there is no $\textsc{Si}(X_i)$. Then X_i in D^k is calculated as

$$X_i^k := \sum_{j=1}^{p} f_j^k(\mathcal{S}_j^k) \ , \tag{3}$$

with $p = 2^{|pa_i^k|}$ and h and $\mathcal{S}$ defined according to our causal model in Sec. 4 of the main text, whereas X_i in D^* is calculated as

$$X_i^* := \sum_{j=1}^{q} f_j^*(\mathcal{S}_j^*) \ , \tag{4}$$

with $q = 2^{|pa_i^*|}$. Under our assumption that the causal model does not change unless an intervention is performed, equations (3) and (4) should be equal and we can therefore write.

$$\sum_{j=1}^{p} f_j^k(\mathcal{S}_j^k) = \sum_{j=1}^{q} f_j^*(\mathcal{S}_j^*) \ , \tag{5}$$

Without loss of generality, we can re-write r.h.s of the equation. (5) as two summations as follows,

$$\sum_{j=1}^{p} f_j^k(\mathcal{S}_j^k) = \sum_{j=1}^{p} f_j^*(\mathcal{S}_j^*) + \sum_{r=p+1}^{q} f_r^*(\mathcal{S}_r^*) \ , \tag{6}$$

where the summation $\sum_{j=1}^{p}$ on both sides of the equation, corresponds to the same indices of the generating functions as well as the same corresponding subset of parents. The summation over r on the r.h.s of eq. (6) contains all the remaining subsets over the power set of pa_i^*. Note that the set of non-linear functions h, over all possible combinations of parents in the power set $\mathcal{P}(pa_i)$ of X_i's parents form a basis and therefore are linearly independent, this implies that the first summation term on the r.h.s is equal to the summation on the l.h.s which in turn implies

$$\sum_{r=p+1}^{q} f_r^*(\mathcal{S}_r^*) = 0 \ .$$

This is possible in one of the two cases: (1) if the basis functions are a linear combination of each other or (2) if the coefficients associated with each of the basis functions is 0. The former we have already ruled out, whereas the latter implies that the coefficients of all the basis $f_r^*(\mathcal{S}_r^*)$ are zero, which implies that there is no edge incoming to X_i in G^* for this set of parents, which is a contradiction.

Next we prove the direction $\textsc{Si}(X_i^k) \longrightarrow \mathrm{pa}_i^k \subset \mathrm{pa}_i^*$ for Lemma 2. Assume that $\textsc{Si}(X_i^k)$ holds, pa_i^k are the actual set of X_i's parents in D^k after $\textsc{Si}(X_i^k)$ but we instead find pa_i' such that $\mathrm{pa}_i' = \mathrm{pa}_i^*$.

Recall that since we are using linear regression, our aim for $X_i \in D^k$ is to minimize

$$\mathbb{E}\left[\left(X_i - \sum_{j=1}^{q} f_j(\mathcal{S}_j)\right)^2\right] \ .$$

Without loss of generality, we can divide the summation term in two parts, the first part consists of the basis containing only pa_i^k and the second part consists of the remaining set of basis.

$$\mathbb{E}\left[\left(X_i - \sum_{j=1}^{p} f_j(\mathcal{S}_j) - \sum_{r=p+1}^{q} f_r(\mathcal{S}_r)\right)^2\right] \ . \tag{7}$$

Since the true generating mechanism for X_i only comprises of basis in the first summation term, we are only left with the noise term ϵ_i associated with X_i. Hence can further simplify eq. (7) to

$$\mathbb{E}\left[\left(\epsilon_i - \sum_{r=p+1}^{q} f_r(\mathcal{S}_r)\right)^2\right] \ . \tag{8}$$

The minimum for eq. (8) is achieved when $\sum_{r=p+1}^{q} f_r(\mathcal{S}_r) = \mathbb{E}(\epsilon_i)$. By our modelling assumptions, we know that $\mathbb{E}(\epsilon_i) = 0$. Therefore, by the same reasoning used to prove reverse direction, we can conclude that the coefficient associated with each of the basis functions in $\sum_{r=p+1}^{q} f_r(\mathcal{S}_r)$ is zero. This implies that $pa_i' \subset pa_i^*$, which is a contradiction. □

Lemma 3 *If $\boldsymbol{\Upsilon}$ is conservative, $\bigcup_{k=1}^{d} G^k = G^*$, If $\boldsymbol{\Upsilon}$ is non-conservative, $\bigcup_{k=1}^{d} G^k \subseteq G^*$.*

Proof: If Υ is conservative, $\forall X_i \in \boldsymbol{X}\ \exists D^k \in \boldsymbol{D}$ such that $pa_i^k = pa_i^*$. We get $\forall X_i\ \bigcup_{k=1}^{d} pa_i^k = pa_i^*$, which implies that $\bigcup_{k=1}^{d} \mathcal{E}(G^k) = \mathcal{E}(G^*)$.

If Υ is non-conservative, $\exists X_i \in \boldsymbol{X}$ such that $\forall D^k \in \boldsymbol{D}\ pa_i^k \subset pa_i^*$. This implies that $\exists X_i\ s.t.\ \bigcup_{k=1}^{d} pa_i^k \subseteq pa_i^*$, which implies that $\bigcup_{k=1}^{d} \mathcal{E}(G^k) \subseteq \mathcal{E}(G^*)$. □

Theorem 4 *Let $\mathcal{Y}$ be the set of all effects such that $\forall Y_i \in \mathcal{Y},\ |pa_i| = 1$. If $\forall Y_i, k\ \alpha_i^k \to 0$, $L(\boldsymbol{D}, M)$ will be the lowest for the true fully-oriented causal network.*

Theorem 5 *Even if the low noise assumption is violated, $L(\boldsymbol{D}, M)$ correctly identifies the collider structures in the underlying causal network.*

As described in the main text, to prove both Thm. 4 and 5, it suffices to show that $L(\boldsymbol{D}, M)$ is a valid L_0 regularized score (e.g. BIC). Note that showing $L(\boldsymbol{D}, M)$ is a valid L_0 regularized score suffices to prove Thm. 5 and the only additional step needed to prove Thm. 4 is the low-noise assumption as this lets us identify any remaining bivariate cases in the resulting Markov Equivalence class (Marx and Vreeken 2019).

Proof: We can write $L(\boldsymbol{D}, M)$ as

$$\begin{aligned} L(\boldsymbol{D}, M) &= L_{str}(M) + \sum_{k=1}^{d} L_{mec}(M^k|M) + \sum_{i=1}^{m} L(\epsilon_i^k) \\ &= L_{str}(M) + \sum_{k=1}^{d}\sum_{i=1}^{m} L(f_i^k) + L(\epsilon_i^k) \end{aligned}$$

Since $L_{str}(M)$ only stores the structure of the global network, which is independent of the number of samples n, therefore it is constant w.r.t n. Hence we get

$$L(\boldsymbol{D}, M) = \mathcal{O}(1) + \sum_{k=1}^{d}\sum_{i=1}^{m} L_F(f_i^k) + L(\epsilon_i^k)\ .$$

Next, let us look at the cost of encoding a specific environment, k which is given as

$$\sum_{i=1}^{m} L(f_i^k) + L(\epsilon_i^k)\ .$$

Encoding Residuals Note that we can rewrite the encoding of the residuals $L(\epsilon)$ as

$$b_k n_k \log \hat{\sigma_k}^2 + \mathcal{O}(1)\ ,$$

where the additive constant is independent of the model.

Encoding Functions Next, we upper bound $L(f)$. We get that $|H| \in \mathcal{O}(\log n)$ from our assumptions. Per hinge we need to encode the number of multiplicative terms $L_{\mathbb{N}}(T_j)$, the function type per term $T_j \log |\mathcal{F}|$, the number of possible assignments from terms to parents $\log \binom{|\mathcal{S}|+T_j-1}{T_j}$ and the parameter vector per hinge $L_p(\theta_j)$. Each parameter vector is constant. Since the number of parents are independent of n_k as they are fixed for a certain network, the number of possible interacting terms T_j is also constant w.r.t. n_k, which means that for large n_k $L_{\mathbb{N}}(T_j)$, $T_j \log |\mathcal{F}|$ (for a finite function class) and $\log \binom{|\mathcal{S}|+T_j-1}{T_j}$ are also constants. In addition, we need to encode the number of hinges for each node, which adds to the constant term. Hence, we can rewrite $L_F(h)$ as

$$c_k \log n_k + \mathcal{O}(1)\ .$$

Combining the residual and function cost for a specific environment, we arrive at

$$b_k n_k \log \hat{\sigma_k}^2 + c_k \log n_k + \mathcal{O}(1)\ .$$

If we set $b_k = 1$ and $c_k = \frac{d_k}{2}$, where d_k is the number of degrees of freedom of the model, we arrive at the BIC score.

Since we compute the same score individually for each environment we can compute the sum over these scores and arrive at

$$L(\boldsymbol{D}, M) = \sum_{k=1}^{d} b_k \log n_k + c_k * n_k \log\left(\sigma_k^2\right) + \mathcal{O}(1)\ .$$

□

B Assumptions

In this section we consolidate the assumptions required for our work into a single list. Assumptions 1-5 are necessary to provide identifiability atleast up to the Markov equivalence class, Assumption 6, allows us to identify the fully directed causal network within the Markov equivalence class, and Assumption 7 is necessary for identifiability of intervention targets.

1. Causal Sufficiency. This means that all the relevant variables are included in the data and there are no unobserved confounders (Pearl 2009). Finding fully oriented causal networks is a challenging task, even under causal sufficiency. Although unlikely to hold in practice, almost every causal discovery method (including our competitors, Ges, Ut-Igsp, CdNod, Lingam, Jci-Pc) require assuming causal sufficiency to obtain identifiability of the underlying causal network.
2. Causal Markov and Faithfulness Condition (Spirtes et al. 2000).
3. Causal relationships between the variables can be modeled by a DAG.
4. Each X_i is related to its parents via non-linear functions and additive independent Gaussian noise term with zero mean and unit variance N, regulated by a scaling factor α_i^k. The causal relationship is formally given as $X_i := f_i^k(\text{pa}_i) + \alpha_i^k \cdot N$. We assume that all the noise terms are jointly independent and that $N \perp\!\!\!\perp \text{pa}_i$ for all X_i in environment k.
5. The number of MARS-hinge functions per variable are upper-bounded by $O(\log n)$ (Mian, Marx, and Vreeken 2021). This assumption is required to show identifiability guarantees entailed by our proposed scores. See Proof of Th. 4 and 5 for details.
6. The noise variance is sufficiently small (Marx and Vreeken 2019; Blöbaum et al. 2018) i.e. $\forall i, k \; in$ Eq.(2) $\alpha_i^k \to 0$.
7. The true underlying causal network G that generates the data only changes as a result of either (i) Hard-Interventions $\text{Hi}(X_j)$; or (ii) inhibiting Soft-Interventions $\text{Si}(X_j)$. An intervention always eliminate atleast one edge in G.

C Pseudocode

Algorithm 2: The Orion Algorithm

```
Input: Datasets $\boldsymbol{D}$ over $\boldsymbol{X}$
Output: Array of causal networks $\boldsymbol{G}$
for $k = 1 \dots d$ do
    $G^k \leftarrow \emptyset$
$\boldsymbol{G} \leftarrow [G^1, \dots G^d]$ repeat
    $\boldsymbol{G} \leftarrow$ FORWARDSEARCH$(\boldsymbol{G}, \boldsymbol{D})$
    $\boldsymbol{G} \leftarrow$ BACKWARDSEARCH$(\boldsymbol{G}, \boldsymbol{D})$
until convergence;
return $\boldsymbol{G}$
```

Algorithm 2 shows the Orion algorithm. Orion greedily adds and removes edges to the global resp. local causal networks such that it reduces $L(\boldsymbol{D}, M)$ most. Orion consists of two phases: forward and backward search, repeated until convergence. As our score is lower-bounded at 0, and we only take steps that reduce our score, Orion is guaranteed to converge. Orion learns a global causal network, and a set containing lists of intervention targets, one for each environment. To obtain the local causal network specific to an environment, we can apply the learned interventions to the predicted global network (Lemm. 1 and 3). Next, we describe the ranking mechanism and the search phases.

Algorithm 3: Score Edge Addition

```
Data: edgeset $\mathcal{E}$ over G
Result: priority queue of edges $Q$
$Q \leftarrow \emptyset$
foreach pair $(u, v) \in \mathcal{E}$ do
    $\psi \leftarrow \delta^{\oplus}(e_{uv}) - \delta^{\oplus}(e_{vu})$
    $Q \leftarrow Q \oplus (e_{uv}, \psi)$
    $Q \leftarrow Q \oplus (e_{vu}, -\psi)$
return $Q$
```

Algorithm 4: Score Edge Removal

```
Data: edgeset $\mathcal{E}$ over $\mathbf{G}$
Result: priority queue of edges $Q$
$Q \leftarrow \emptyset$
foreach pair $(u, v) \in \mathcal{E}$ do
    $\psi \leftarrow \delta^{\ominus}(e_{uv})$
    $Q \leftarrow Q \oplus (e_{uv}, \psi)$
return $Q$
```

Edge gain To calculate the gain provided by each edge, we first measure the bits that we save by adding an edge in the current model. Formally, let $e_{ij} = X_i \rightarrow X_j$, and M be the current model. We write $M \oplus e_{ij}$ to denote the model with edge e_{ij} included. We define the absolute gain in bits δ associated with edge e_{ij} as

$$\delta(e_{ij}) = \max \{0, L(\boldsymbol{D}, M) - L(\boldsymbol{D}, M \oplus e_{ij})\}$$

where $L(\boldsymbol{D}, M)$ is the score defined in Sec. 4. Next, we calculate the true gain for this edge by calculating the relative bits we gain over adding this edge in the opposite direction. Formally,

$$\psi(e_{ij}) = \delta(e_{ij}) - \delta(e_{ji}) \ .$$

Intuitively, the higher the value of $\psi(e_{ij})$, the more certain we are that we inferred the correct direction for this edge. This is motivated by the no-hypercompression inequality (Grünwald 2007), which we use to test the significance of each edge. Let $s = \psi(e)$, the probability of gaining s bits over the null model is less than or equal to 2^{-s}. If we find that the gain for an edge is not significant— i.e. 2^{-s} is greater than the desired significance threshold— we do not add this edge to the network. For the backward search the ranking, $\delta^{\ominus}$, is analogous to forward search except that the gain for each edge is the number of bits saved by removing the edge. Algorithms 3 and 4 show the pseudocode for edge scoring for both edge addition and removal.

Algorithm 5: Forward Search

```
Data: Environments $\boldsymbol{D}$ over $\boldsymbol{X}$, array of causal networks $\mathbf{G}$
Result: Array of updated networks $\mathbf{G}$
$\mathcal{E}^*(\mathbf{G}) \leftarrow$ all possible edges in $\mathbf{G}$
$\mathcal{E}_{cand} \leftarrow \mathcal{E}^*(\mathbf{G}) - \mathcal{E}(\mathbf{G})$
$Q \leftarrow$ ScoreEdgeAddition$(\mathcal{E}_{cand})$
while $Q$ not empty do
    $e \leftarrow$ take top most entry from $Q$
    $ch_e \leftarrow$ child variable for edge $e$
    if $\mathbf{G} \oplus e$ not cyclic and $e$ is significant then
        $\mathbf{G} \leftarrow \mathbf{G} \oplus e$
        foreach edge $e^k$, connected to $ch_e \in Q$ do
            update score of $e^k \in Q$ to $\psi(e^k)$
return $\mathbf{G}$
```

Forward Search In forward search, ORION maintains a priority queue containing the edges ordered by their relative gain,Ψ. We iteratively build the causal graph by adding the highest ranked edge from the priority queue to the causal DAG. We reject edges that introduce cycles in the network. Once an edge e_{ij} is added to the network, we re-rank all the candidate edges associated with variables X_j in the priority queue. This is repeated until all the candidate edges have been evaluated.

We introduce each edge as a part of the global network and therefore each of the datasets only need to pay a discounted cost of storing the causal mechanism in order to include this edge. If this cost is not enough for a local dataset to register a gain, it rejects the edge. In that case, an intervention is encoded for the latter. The pseudocode for forward search is shown in Algorithm 5

Backward Search Since we greedily add edges during the forward search phase, some of the parents of a variable may become redundant as forward search progresses. To remove such edges we need the backward search. We iteratively remove that edge from the network which improves our score the most. We keep removing the edges iteratively, until no edge removal improves the score anymore. We show the psuedocode for Backward search in Algorithm 6.

Algorithm 6: Backward Search

```
Data: Environments $\boldsymbol{D}$ over $\boldsymbol{X}$, array of causal networks $\mathbf{G}$
Result: Array of updated networks $\mathbf{G}$
$\mathcal{E}_{cand} \leftarrow \mathcal{E}(\mathbf{G})$
$Q \leftarrow$ ScoreEdgeRemoval$(\mathcal{E}_{cand})$
while $Q$ not $empty$ do
    $e \leftarrow$ take top most entry from $Q$
    if $e$ is significant then
        $\mathbf{G} \leftarrow \mathbf{G} \ominus e$
        foreach edge $e^k$, connected to $ch_e \in \mathbf{G}$ do
            update score of $e^k \in Q$ to $\psi(e^k)$
return $\mathbf{G}$
```

D Experiments and Additional Results

Competitor Methods The code for the competitor methods were taken from the following sources.

- CdNod: Implementation provided by the authors on Github: https://bit.ly/3m1jo0G.
- Pc (using the Rcit (Strobl, Zhang, and Visweswaran 2019) as the independence test) and Ges: Causal Discovery Toolbox (Kalainathan and Goudet 2019).
- LinGAM: Implementation provided by the authors on Github: https://bit.ly/39PExra.
- Jci-Fci: Implementation provided by the authors on Github: https://bit.ly/3MNL6Ju.
- Ut-Igsp: Implementation provided by the authors on Github: https://bit.ly/3tVbHO1.

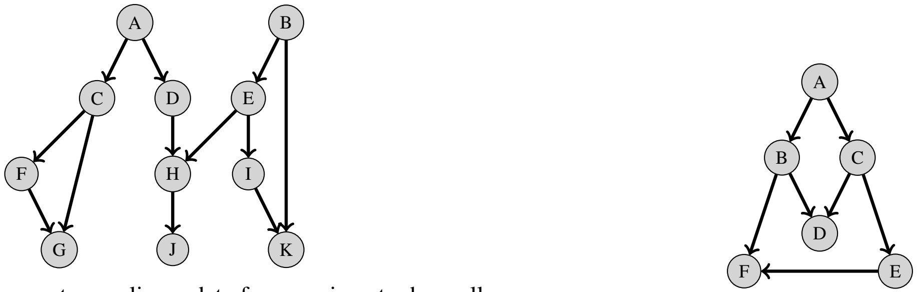
Comparisons to LinGAM The original LinGAM is not applicable to multi-environment setting. We, however, compare to the Direct-LiNGAM (Shimizu 2012) that is specifically designed for multi-environment setting. Direct-LiNGAM returns one weight-matrix per dataset where non-zero entries correspond to a causal edge. Using these individual local causal networks in conjunction with Lemmas 1 and 3, we can find the global causal network as well as the intervention targets exactly as we do for Orion.

Comparisons to Ges For baseline FGes we simply learn individual local networks and take the union as the predicted causal network, which is guaranteed to be correct in the limit. Once we have individual local causal networks, using Lemmas 1 and 3 we can find the global causal network as well as the intervention targets exactly as we do for Orion. The second variant, EGes, is designed for multi-environment setting. It consists of context variables that we can directly use to identify intervention targets, similar to the Jci framework.

Instantiation We implement Orion in Python, and use the nonparametric regression splines from the open source R-package, Earth. We further standardize all data to avoid susceptibility to practical issues like var-sortability (Reisach, Seiler, and Weichwald 2021). Orion is implemented to work in parallel over each environment. We run all experiments on an 3.5 Ghz Intel E5-2643 CPU. For all input instances, Orion inferred causal network inside 3 minutes for parallelized version and within 20 minutes for single-threaded runs.

Evaluation Metrics We evaluate the discovered causal networks in terms of structural similarity using the Structural Hamming Distance (Shd) (Kalisch and Bühlmann 2007). Let G and H be the ground truth resp. predicted causal DAG, then Shd(G, H) counts the edges where the two causal DAGs differ. Structural similarity alone, however, is not enough since a single wrongly oriented causal edge can lead to more than one incorrect causal decisions. Therefore it is critical that we also measure the causal similarity between causal networks. We do this using the Structural Intervention Distance (Sid) (Peters and Bühlmann 2015). Sid(G, H) counts the pairs of variables X_i and X_j, such that the effect of an intervention on X_i is incorrectly propagated to X_j. For methods that output the Markov equivalence class of the causal network, Sid is an interval indicating the best resp. worst possible score over this class.

For comparability across different settings, we normalize both Shd and Sid between 0 and 1. The minimum for both Shd and Sid is zero. The maximum value of Shd for m different variables is if all possible edges in a network are incorrectly predicted by the algorithm, there are a total of $\binom{m}{2}$ edges possible for m variables. The maximum Sid for m variables is given by $m * (m-1)$ (Peters and Bühlmann 2015). Once we have these minimum and maximum values, we can normalize any value of Shd resp. Sid between 0 and 1.



(a) DAG used to generate non-linear data for experiment where all data are interventional. This graph contains all connections possible in a DAG: a collider ($D \rightarrow H \leftarrow E$), a fork ($C \leftarrow A \rightarrow D$), a chain ($A \rightarrow C \rightarrow F$) and a feed forward loop ($C \rightarrow F \rightarrow G$ and $C \rightarrow G$).

(b) DAG used to generate non-linear data for the experiments involving intervention targets detection.

D.1 Specific Experimental Setup

Q1. Can ORION discover causal networks over data from multiple environments? We have number of environments $d \in \{3, 5, 7, 9\}$, number of variables $m \in \{5, 10, 15\}$ and number of samples per environment $n = 1000$ as our experimental setting. We simulate DAGs using the Erdos-Rényi model. We model effect as a function of its causes using polynomial functions in half of the cases. For other half we use randomly initialized. 2-layer neural networks to model the mechanism. To generate synthetic data, we use the causal discovery toolbox (Kalainathan and Goudet 2019). The full set of results for this setting are provided in Tables. 2 and 3.

Data for the case where each environment comes from a different interventioned (sub)network is generated using the graph structure shown in Figure 4a. For each environment we randomly pick a variable and apply either an Inhibiting or a Hard intervention on the latter with equal probability. We use $n = 3000$ samples per environment.

Q2. How well does ORION perform on networks where our assumptions may not hold? We use the re-simulated Lung-cancer gene expression, REGED network (Statnikov et al. 2015), containing 500 samples from http://www.causality.inf.ethz.ch/challenge.php?page=datasets. We extract two non-overlapping connected components of 5 resp. 15 variables which we refer to REGED5b resp. REGED15. Next, we break our assumptions by introducing selection bias. We sort each dataset once on each variable and divide the resulting dataset into three overlapping environments which gives us 5 resp. 15 distinct datasets. We refer to these datasets as REGED5s resp. REGED15s. We report the results for these experiments in Fig **??**.

Q3. Can ORION reliably identify intervention targets? Data for the case where we identify the intervention targets is generated using the graph structure that is shown in Figure 4b. This is the same structure as used by (Zhang et al. 2017) for *all* of their synthetic data experiments to test their implementation of CDNOD. We use the settings $d \in \{3, 5, 7\}$ for the number of environments and use $n = 3000$ samples per environment. A total of 100 experimental instances are generated. We report the results for identifying only the direct intervention targets in Fig. 5.

D.2 Additional Results

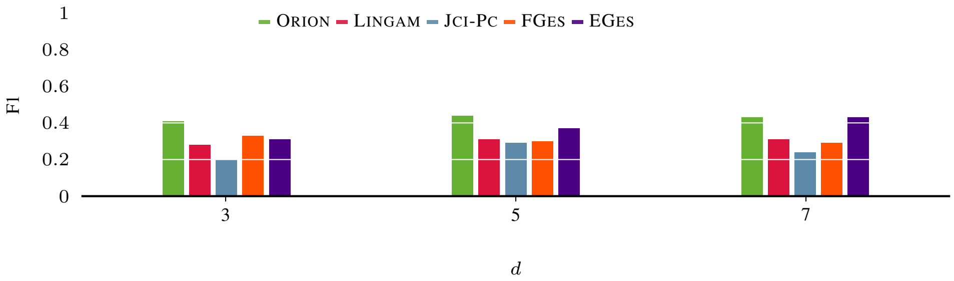


Figure 5: [Higher is better] F1 scores for ORION, LINGAM, JCI-PC and GES for identifying direct intervention targets in synthetic data over different environment sizes, d.

d	m	Orion	Lingam	Jci-Pc	CdNod-inv	Ut-Igsp	Jci-Fci	EGes
3		3.2	5.3	3.4	3.5	3.4	4.9	4.3
5	**5**	3.5	5.7	4.0	3.3	3.5	4.9	4.2
7		3.4	5.8	4.1	3.5	3.6	4.9	4.4
9		3.4	6.1	5.7	3.3	3.6	4.8	4.6
3		18.0	24.8	16.6	15.7	20.2	19.3	21.1
5	**10**	18.0	26.1	16.6	14.8	19.8	19.3	22.0
7		18.0	27.4	17.1	18.0	20.2	19.3	22.9
9		17.5	27.9	18.4	21.1	20.2	19.8	23.8
3		40.1	58.8	39.9	37.8	56.7	45.1	60.9
5	**15**	40.1	61.9	39.9	34.6	55.6	45.1	60.9
7		40.1	65.1	39.9	31.5	56.7	45.1	60.9
9		42.0	67.2	37.8	33.6	55.6	45.1	60.9

Table 2: [Lower is better] Averaged Shd for experiments involving homogeneous synthetic data.

d	m	Orion	Lingam	Jci-Pc	CdNod-inv	Ut-Igsp	Jci-Fci	EGes
3		5.2	9.4	[3.2, 9.2]	[3.2,9.6]	7.2	9.6	[4.0,8.4]
5	**5**	5.2	9.2	[3.2,10.2]	[3.4,8.8]	7.4	9.6	[3.8,8.0]
7		5.0	9.0	[2.8, 9.2]	[5.6,8.6]	7.2	9.6	[4.4,8.6]
9		4.8	9.2	[3.4,11.2]	[5.8,8.4]	7.6	9.6	[4.2,8.8]
3		40.5	52.2	[42.3,60.3]	[40.5,51.3]	52.2	58.5	[36.0,46.8]
5	**10**	39.6	49.5	[40.5,60.3]	[32.4,40.5]	52.2	59.4	[36.9,45.9]
7		37.8	47.7	[37.8,58.5]	[50.4,59.4]	51.3	59.4	[38.7,46.8]
9		38.7	46.8	[39.6,56.7]	[48.6,58.5]	51.3	60.3	[37.8,45.9]
3		113.4	147.0	[134.4,168.0]	[123.9,134.4]	155.4	163.8	[130.2,138.6]
5	**15**	111.3	138.6	[134.4,161.7]	[119.7,130.2]	153.3	161.7	[126.0,134.4]
7		109.2	132.3	[132.3,163.8]	[136.5,144.9]	155.4	161.7	[123.9,132.3]
9		109.2	132.3	[130.2,157.5]	[123.9,142.8]	153.3	159.6	[128.1,136.5]

Table 3: [Lower is Better] Averaged Sid for experiments involving homogeneous synthetic data. Intervals indicates the best, resp. worst possible intervention distance for methods that output the Markov equivalence class of the causal network.