Budgeted Experiment Design for Causal Structure Learning

AmirEmad Ghassami†, Saber Salehkaleybar†, Negar Kiyavash†, Elias Bareinboim†

†Department of ECE, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA.
†Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA.
†Department of Computer Science, Purdue University, West Lafayette, Indiana, USA.
†{ghassam2,sabersk,kiyavash}@illinois.edu, †eb@purdue.edu

Abstract

We study the problem of causal structure learning when the experimenter is limited to perform at most $k$ non-adaptive experiments of size 1. We formulate the problem of finding the best intervention target set as an optimization problem, which aims to maximize the average number of edges whose directions are resolved. We prove that the objective function is sub-modular and a greedy algorithm is a $(1 - \frac{1}{e})$-approximation algorithm for the problem. We further present an accelerated variant of the greedy algorithm, which can lead to orders of magnitude performance speedup. We validate our proposed approach on synthetic and real graphs. The results show that compared to the purely observational setting, our algorithm orients majority of the edges through only a small number of interventions.

Introduction

The problem of learning the causal relations among a set of variables is of interest in many scientific domains and AI. Causal structures are commonly represented by directed acyclic graphs (DAGs), where the vertices are random variables and a directed edge from variable $X$ to $Y$ indicates that variable $X$ is a direct cause of $Y$ (Pearl 2009; Spirtes, Glymour, and Scheines 2000). In order to uncover the causal relations among a set of variables, if the experimenter is restricted to work with only observational data from the variables, by running conditional independence based algorithms, he is usually left with some (or in some cases many) causal relations unresolved. On the other hand, sufficient experiments involving interventions can identify the underlying causal graph completely.

An interventional inference algorithm consists of a set of experiments. In each experiment, the experimenter intervenes on a subset of the variables and collects data from the intervened system. In this setting, two natural questions arise:

1. What is the smallest required number of experiments in order to fully learn the causal graph?
2. For a fixed number of experiments (budget), what portion of the causal graph can be learned?

The first problem has been addressed in the literature under different assumptions (see Related Work section), while the second question has received less attention. We address the second question herein. Specifically, we consider a setup with $k$ experiments, each containing exactly one intervention. The reason we consider single-intervention experiments is that in many applications, such as some experiments in biology, performing simultaneous interventions in multiple variables may not be feasible.

The majority of the work in the field of causality separates the design of algorithms from dealing with the statistical concerns. The reason is that the statistical tests required are often well-studied in statistics. Many of the seminal works in the field of causal inference (Verma and Pearl 1991; Pearl and Bareinboim 2011; Shpitser and Pearl 2006; Shpitser, VanderWeele, and Robins 2012; Hyttinen, Eberhardt, and Hoyer 2013; Eberhardt, Glymour, and Scheines 2006; Schölkopf et al. 2012; Hoyer et al. 2009), and almost entire books (Spirtes, Glymour, and Scheines 2000) and (Pearl 2009) decouple the statistical from the algorithmic considerations. We focus on the algorithm design side of the problem and formulate it as an optimization problem. We propose an approximation algorithm which allows an experimenter to take high advantage of a limited budget of experiments.

Contributions. In our interventional inference algorithm, first an observational test, such as PC algorithm (Spirtes, Glymour, and Scheines 2000), is performed on the set of variables. This test learns the skeleton as well as the orientation of some of the edges of the causal graph. Based on the result of the initial stage, the complete set of $k$ experiments is designed in a non-adaptive manner. The advantage of a non-adaptive approach is that having the complete set of experiments enables the experimenter to perform the interventional experiments in parallel. The formal description of the problem of interest is provided in the Problem Description Section. We present the problem of finding the best intervention target set as an optimization problem which aims to maximize the average number of edges whose directions are resolved. We prove that this objective function is monotonically increasing and sub-modular. This implies that a general greedy algorithm is a $(1 - \frac{1}{e})$-approximation algorithm. Since computing the objective function is intractable for a given set, we present an unbiased estimation of this function. Additionally, we propose a fast estimator with a close approximation.
Performance to the unbiased estimator (Proposed Approach Section). In Improved Algorithm Section, we implement an accelerated variant of the general greedy algorithm through lazy evaluations, originally proposed by Minoux [Minoux 1978]. This algorithm can lead to orders of magnitude performance speedup (Leskovec et al. 2007). Using synthetic and real data, in Experiments Section, we show that the proposed approach recovers a significant portion of the edges by only a few number of interventions in many graphs of interest.

Related Work
The generic algorithms for purely observational setting are IC and IC* (Pearl 2009), PC and FCI (Spirtes, Glymour, and Scheines 2000 and Greedy Equivalence Search (GES) algorithm (Meek 1997, Chickering 2002). Such purely observational approaches reconstruct the causal graph up to the Markov equivalence class, and hence, the direction of some of the edges may remain unresolved. Albeit under some stronger assumptions, full structure learning using merely observational data is feasible (Shimizu et al. 2006, Hoyer et al. 2009, Peters and Bühlmann 2012). There is a growing body of research on learning causal structures using interventional data (Eberhardt 2007, Hauser and Bühlmann 2012, Cooper and Yoo 1999, He and Geng 2008). Specifically, Hauser and Bühlmann (Hauser and Bühlmann 2012) extended the notion of Markov equivalence to the interventional case and proposed an interventional generalization of GES algorithm. Note that performing an intervention could be viewed as optimally designing a change in the environment. In (Peters, Bühlmann, and Meinshausen 2016, Ghassami et al. 2017) the case where data comes from different unknown changes in the environment is studied.

Regarding the first question discussed earlier, (Eberhardt, Glymour, and Scheines 2005) considered the complete graph as the underlying causal structure to obtain the worst case bounds on the number of required experiments under the assumption of causal sufficiency. In that work, both cases of experiments containing bounded and unbounded number of interventions were studied. In (Eberhardt 2007, Hyttinen, Eberhardt, and Hoyer 2013), it was shown that the problem of designing a proper set of experiments for causal inference is connected to the problem of finding a separating system in a graph. Hence, through this connection, the authors derived fundamental bounds on the number of required experiments for fully learning the graph.

Adaptive algorithms for experiment design were proposed in (Hauser and Bühlmann 2014, Shanmugam et al. 2015). In (Hauser and Bühlmann 2014), two algorithms that minimize the number of experiments in the worst case were proposed. These algorithms are adaptive, and in the one with polynomial complexity, the size of experiments can be as large as half the order of the graph, which may not be practical in many real-life applications. In (Shanmugam et al. 2015), the authors present information-theoretic lower bounds on the number of required experiments for both deterministic and randomized adaptive approaches. They also proposed an adaptive algorithm that allowed to learn chordal graphs completely. In (Kocaoglu, Dimakis, and Vishwanath 2017), the authors considered costs for intervening on each variable and derived an experiment design algorithm with minimum total cost that reconstructs the whole structure. They also considered the case in which the number of experiments is limited. However, each experiment in their setup is allowed to include intervening on arbitrary number of variables.

The budgeted experiment design problem is similar to the influence maximization problem. The goal in the latter is to find k vertices (seeds) in a given network such that under a specified influence model, the expected number of vertices influenced by the seeds is maximized. In (Kempe, Kleinberg, and Tardos 2003) the authors showed that selecting the most influential vertices is NP-hard and provided the first provable approximation guarantees for efficient algorithms based on submodularity of the objective function. There are many improvements of (Kempe, Kleinberg, and Tardos 2003) such as (Leskovec et al. 2007) and (Chen, Wang, and Yang 2009). Besides the interpretative differences, an important distinction between the two problems is that in maximum influence problem, the goal is to spread the influence to the vertices of the graph while in budgeted experiment design problem, the goal is to pick the initial k vertices in a way that leads to discovering the orientation of as many edges as possible. As such the optimal solution to these two problems for a given graph can be quite different (see the appendix for an example).

Problem Description
Definition 1. A mixed graph $G = (V, E, A)$ consists of set of vertices $V$, set of undirected edges $E$, and set of directed edges $A$. A directed graph $G = (V, A)$ consists of vertex set $V$ and set of directed edges $A$. Directed graph $G$ is a DAG if it is a finite graph with no directed cycles.

A DAG $G$ is called causal if its vertices represent random variables $V = \{X_1, ..., X_n\}$ and a directed edge $(X_i, X_j)$ indicates that variable $X_i$ is a direct cause of variable $X_j$. In our model, we assume no restrictions on the underlying causal DAG. We consider a structural equation model (Pearl 2009) on the variables, which is a collection of $n$ equations $X_i = f_i(PAX_i, N_i), i \in \{1, ..., n\}$, where $PAX_i$ denotes the parents of $X_i$ in $G$, and $N_i$ are jointly independent noise variables. Furthermore, we assume faithfulness assumption on the probability distribution $P_V$ and causal sufficiency (Spirtes, Glymour, and Scheines 2000, Pearl 2009).

Definition 2. Two causal DAGs $G_1$ and $G_2$ over $V$ are Markov equivalent if every distribution that is compatible with one of the graphs is also compatible with the other. The set of all graphs over $V$ is partitioned into a set of mutually exclusive and exhaustive Markov equivalence classes, which are the set of equivalence classes induced by the Markov equivalence relation (Koller and Friedman 2009). The graph union of all DAGs in the Markov equivalence class of a $\text{Graph union}[\text{of all DAGs in the Markov equivalence class of a}]}$
DAG $G$ is called the essential graph of $G$ and is denoted by $\text{Ess}(G)$.

Using purely observational data (referred to as the null experiment by [Eberhardt, Glymour, and Scheines 2005]), one can learn the causal structure up to Markov equivalence. This could be done by performing a “complete” conditional independence based algorithm which we define as follows.

**Definition 3.** A complete conditional independence-based (CCI) algorithm includes first finding the skeleton and v-structures of the graph by performing conditional independence tests on the given observational data, and then, applying the Meek rules.

A CCI algorithm leads to learning the Markov equivalence class of the ground truth DAG. After performing CCI on the ground truth DAG $G^*$, we denote the set of directed edges revealed by the procedure as $A(\text{Ess}(G^*))$ and the set of undirected edges by $E(\text{Ess}(G^*))$.

Interventions can enable us to differentiate among the different causal structures within a Markov equivalence class. We use the same notion of ideal intervention as in [Eberhardt, Glymour, and Scheines 2005]. That is, for an intervention $I$ on variable $X \in V$, the influence of all the variables on the target variable $X$ is removed, and $X$ is randomized by forcing values from an independent distribution on it. This intervention changes the joint distribution of all variables in the system for which $X$ is a direct or indirect cause. We denote this intervention by $I = X$. An interventional inference algorithm consists of a set of $k$ experiments, $\mathcal{E} = \{E_1, E_2, \ldots, E_k\}$, where each experiment $E_i$ contains $m_i$ interventions, i.e., $E_i = \{I^{(i)}_1, I^{(i)}_2, \ldots, I^{(i)}_{m_i}\}$ for $1 \leq i \leq k$.

An intervention algorithm may be adaptive, in which the experiments are performed sequentially and the information obtained from the previous experiments is used to design the next one, or passive, in which all the experiments are designed beforehand.

The approach we take in our algorithm is to first perform a CCI algorithm to learn the skeleton and the direction of the edges in $A(\text{Ess}(G^*))$, and then design the experiments in a passive manner under a budget constraint. This approach gives the experimenter the ability to perform the experiments in parallel without the need to wait for the result of one experiment to choose the next one. For example, in the study of gene regulatory networks (GRNs), for the cases that the GRN of all cells are the same, experiments can be performed on different cells simultaneously.

We consider a setup in which a budget of $k$ interventions is given and the goal is to design a set of experiments to learn the directions of as many edges in $E(\text{Ess}(G^*))$ as possible.

We focus on single-intervention experiments setup in which for all experiments, we have $m_i = 1$. Therefore, our experiment set is of the form $\mathcal{I} = \{I_1, I_2, \ldots, I_k\}$. To simplify the notation, we donate the set of intervention targets as $\mathcal{I} = \{I_1, I_2, \ldots, I_k\}$, where $I_i = I_1^{(i)}$. Moreover, as shown in [Eberhardt 2007], observing the result of the null experiment, one can find the orientation of the edge $(X_i, X_j) \in E(\text{Ess}(G^*))$, if there exists $E_i \in \mathcal{E}$ such that $(X_i \in E_i, X_j \notin E_i)$ or $(X_j \in E_i, X_i \notin E_i)$. On the other hand, if for all experiments $E_i \in \mathcal{E}$ we have both $X_i \in E_i$ and $X_j \in E_i$, the orientation of $(X_i, X_j)$ cannot be learned. An experiment in which both $X_i$ and $X_j$ are intervened on is called a zero information experiment for $X_i$ and $X_j$. Having $m_i = 1$, for all $i \in \{1, \ldots, k\}$, prevents such zero information experiments.

So formally, we focus on the following problem: \textit{If the experimenter is allowed to perform $k$ experiments, each of size 1, what portion of the graph could be reconstructed on average?} We formalize the problem statement as follows. For any given set of directed edges $A$ from the underlying DAG $G^*$, define $R(A, G^*)$ as the set of edges which do not belong to $E(\text{Ess}(G^*))$ and their direction can be learned by applying Meek rules starting from the set of directed edges $A \cup E(\text{Ess}(G^*))$; i.e., the set of edges whose direction is learned as a result of new information in $A$. As mentioned earlier, intervening only on variables $\mathcal{I} \subseteq V$, we learn the orientation of all the edges intersecting with members of $\mathcal{I}$. Denoting the set of these edges by $A_G^{(\mathcal{I})}$, intervening on variables in $\mathcal{I}$ results in learning the direction of edges in $R(A_G^{(\mathcal{I})}, G^*)$. That is, $R(A_G^{(\mathcal{I})}, G^*)$ is the set of edges whose direction is learned as a result of intervening on $\mathcal{I}$ and applying Meek rules on $A_G^{(\mathcal{I})}$. Let $D(\mathcal{I}, G^*) = |R(A_G^{(\mathcal{I})}, G^*)|$, and consider $\mathcal{G}$ as the set of all DAGs in the Markov equivalence class of $G^*$. Since we do not know the ground truth we define the average number of edges whose direction can be learned as a result of intervention on target set $\mathcal{I}$ as

$$D(\mathcal{I}) = \frac{1}{|\mathcal{G}|} \sum_{G_i \in \mathcal{G}} D(\mathcal{I}, G_i).$$

Therefore, the problem of interest could be stated as finding intervention target set $\mathcal{I} \subseteq V$ of cardinality $k$ which maximizes $D(\mathcal{I})$, that is

$$\max_{\mathcal{I} \subseteq V} D(\mathcal{I}) \quad \text{s.t.} \quad |\mathcal{I}| = k. \quad (1)$$

This problem is hard to solve for two reasons: First, finding the optimum $\mathcal{I}$ requires a combinatorial search. Second, even for a given set $\mathcal{I}$, computing $D(\mathcal{I})$ requires considering all the DAGs in the Markov equivalence class of $G^*$, which in the worst case, say, if $G^*$ is a complete graph, requires super-exponential computation time.

**Proposed Approach**

**Definition 4.** A set function $f : 2^V \rightarrow \mathbb{R}$ is monotonically increasing if for all sets $\mathcal{I}_1 \subseteq \mathcal{I}_2 \subseteq V$, we have $f(\mathcal{I}_1) \leq f(\mathcal{I}_2)$.
Definition 5. A set function \( f : 2^V \rightarrow \mathbb{R} \) is submodular if for all subsets \( I_1 \subseteq I_2 \subseteq V \) and all \( v \in V \setminus I_2 \)

\[
f(I_1 \cup \{v\}) - f(I_1) \geq f(I_2 \cup \{v\}) - f(I_2).
\]

In their seminal work, Nemhauser et. al. showed that if \( f \) is a submodular, monotone non-decreasing set function with \( f(\emptyset) = 0 \), then the set \( \mathcal{I} \) with \( |\mathcal{I}| = k \) found by the greedy algorithm satisfies \( f(\mathcal{I}) \geq \left(1 - \frac{1}{e}\right) \max_{|\mathcal{I}|=k} f(\mathcal{I}) \) (Nemhauser, Wolsey, and Fisher 1978). That is, the greedy algorithm is a \( (1 - \frac{1}{e}) \)-approximation algorithm. We will use this result in our proposed approach. We will show that the set function \( D \) is monotonically increasing and submodular, and hence, the greedy algorithm is a \((1 - \frac{1}{e})\)-approximation algorithm to the maximization problem \([\mathcal{I}]\).

Lemma 1. The set function \( D \) is monotonically increasing, i.e., for sets \( I_1 \subseteq I_2 \), we have

\[
D(I_2) \leq D(I_1).
\]

Proof. First we show that for a given directed graph \( G_i \in \mathcal{G} \) the function \( D(I, G_i) \) is a monotonically increasing function of \( I \). In the proposed method, intervening on elements of \( I \), we first discover the orientation of the edges in \( A(G_i) \), and then applying the Meek rules, we possibly learn the orientation of some extra edges. Having \( I_1 \subseteq I_2 \) implies that \( A(G_{I_1}) \subseteq A(G_{I_2}) \). Therefore using \( I_2 \), we have more information about the direction of edges. Hence, in the step of applying Meek rules, by soundness and order-independence of Meek algorithm, we recover the direction of more extra edges, i.e., \( R(A(G_{I_1}), G_i) \subseteq R(A(G_{I_2}), G_i) \), which in turn implies that \( D(I_1, G_i) \leq D(I_2, G_i) \). Finally, from the relation \( D(I) = \frac{1}{|\mathcal{G}|} \sum_{G_i \in \mathcal{G}} D(I, G_i) \), the desired result is immediate.

The following lemma plays a fundamental role in the proof of submodularity of the set function \( D \).

Lemma 2. For sets \( I_1, I_2 \subseteq V \),

\[
R(A(G_{I_1 \cup I_2}), G^*) = R(A(G_{I_1}), G^*) \cup R(A(G_{I_2}), G^*).
\]

See the appendix for the proof.

Theorem 1. The set function \( D \) is a supermodular function.

Proof. Due to Lemma 1 it suffices to show that for \( I_1 \subseteq I_2 \subseteq V \), and \( v \in V \), we have \( D(I_1 \cup \{v\}) - D(I_1) \geq D(I_2 \cup \{v\}) - D(I_2) \). First we show that for a given directed graph \( G_{I_1} \in \mathcal{G} \) the function \( D(I, G_{I_1}) \) is a submodular function of \( I \). From Lemma 2 we have \( R(A(G_{I_1 \cup \{v\}}), G_i) = R(A(G_{I_1}), G_i) \cup R(A(G_{\{v\}}), G_i) \). Therefore,

\[
D(I_1 \cup \{v\}, G_i) - D(I_1, G_i) = |R(A(G_{I_1 \cup \{v\}}), G_i)| - |R(A(G_{I_1}), G_i)|
\]

Similarly, \( D(I_2 \cup \{v\}, G_i) - D(I_2, G_i) = |R(A(G_{I_2 \cup \{v\}}), G_i)| - |R(A(G_{I_2}), G_i)| \). Since \( I_1 \subseteq I_2 \), as seen in the proof of Lemma 1, \( R(A(G_{I_1}), G_i) \subseteq R(A(G_{I_2}), G_i) \). Therefore, \( |R(A(G_{I_1}), G_i)| \leq |R(A(G_{I_2}), G_i)| \), which implies that \( D(I_1 \cup \{v\}, G_i) - D(I_2, G_i) \geq D(I_2 \cup \{v\}, G_i) - D(I_2, G_i) \). This together with the fact that the function \( D(I, G_i) \) is a monotonically increasing function of \( I \) (observed in the proof of Lemma 1) shows that \( D(I, G_i) \) is a submodular function of \( I \). Finally, we have \( D(I) = \frac{1}{|\mathcal{G}|} \sum_{G_i \in \mathcal{G}} D(I, G_i) \). Since a non-negative linear combination of submodular functions is also submodular, the proof is concluded.

As mentioned in Problem Description Section, another regarding solving the optimization problem \([\mathcal{I}]\) is the computational intractability of calculating \( D(I) \) for a given intervention target set \( I \). We propose running Monte-Carlo simulations of the intervention model for sufficient number of times to obtain an accurate estimation of \( D(I) \). The pseudo-code of a baseline estimator is presented in Subroutine 1. In this subroutine, we first orient the edges in \( E(\text{Ess}(G^*)) \) in one of the possible directions independently according to Bernoulli(\( \frac{1}{2} \)) distribution to obtain a directed graph \( G' \). We keep those DAGs which are in the same Markov equivalence class as \( G^* \) in a multiset \( \mathcal{G}' \).

To check whether \( G' \) is in the same Markov equivalence class as \( G^* \), if suffices to check if they have the same set of \( v \)-structures (Verma and Pearl 1991). Note that \( G' \) is a multiset in which repetition is allowed, and operator \( \cup \) in the pseudo-code indicates the multiset addition. Finally, we calculate the estimated value \( \bar{D}(I) \) on \( G' \) instead of \( G \) as

\[
\bar{D}(I) = \frac{1}{|\mathcal{G}'|} \sum_{G_i' \in \mathcal{G}'} D(I, G_i').
\]

Theorem 2. For any \( I \subseteq V \), \( \bar{D}(I) \) obtained from Subroutine 1 is an unbiased estimate of \( D(I) \), i.e., \( \mathbb{E}[\bar{D}(I)] = D(I) \).

See the appendix for the proof.

Our experiments showed that in some cases, out of \( T \) generated graphs in Subroutine 1, only a few of them are in the same Markov equivalence class as \( G^* \). Therefore, we propose an efficient estimator for \( D(I) \). The pseudo-code of the proposed estimator is presented in Subroutine 2. In this subroutine for the given mixed graph \( E(\text{Ess}(G^*)) \), we generate \( N \) DAGs from the Markov equivalence class of \( G^* \) as follows: We consider all subsets of size 3 from \( V \) in a random uniformly random order (achieved by uniformly shuffling the labels of elements of \( V \)). For each subset \( \{X_i, X_j, X_k\} \), we orient the undirected edges among \( \{X_i, X_j, X_k\} \) independently according to Bernoulli(\( \frac{1}{2} \)) distribution. If the resulting orientation on the induced subgraph on \( \{X_i, X_j, X_k\} \) became a directed cycle or a new \( v \)-structure which was not in \( \text{Ess}(G^*) \), we redo the orienting. We keep checking all the subsets of size 3 until the induced subgraph on all of them are directed and none of them
**Subroutine 1** Baseline $D(\mathcal{I})$ Estimator Subroutine

**Input:** $\mathcal{Ess}(G^*)$, intervention target set $\mathcal{I}$, and $T$.

**initiate:** $G^0 = \emptyset$, $i = 1$

for $j = 1$ to $T$ do

Orient each undirected edge $\{u, v\} \in E(\mathcal{Ess}(G^*))$ as $(u, v)$ or $(v, u)$ independently according to $\text{Bern}(\frac{1}{2})$ to get directed graph $G^i$.

if $G^i$ is a DAG and is in the Markov equivalence class of $G^*$ then

$G^i = G^*$

$G^i = G^i \cup G^i$

end if

end for

Output: $\hat{D}(\mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{G^i \in \mathcal{G}^i} |D(\mathcal{I}, G^i)|$

---

**Subroutine 2** $D(\mathcal{I})$ Estimator Subroutine

**Input:** $\mathcal{Ess}(G^*)$, intervention target set $\mathcal{I}$, and $N$.

**initiate:** $G^0 = \emptyset$

for $i = 1$ to $N$, generate $G^i$ as follows: do

Uniformly shuffle the order of the elements of $V$.

while the induced subgraph on any subset of size 3 of the variables is not directed, or a directed cycle, or a $v$-structure which was not in $\mathcal{Ess}(G^*)$ do

for all $\{X_i, X_j, X_k\} \subseteq V$ do

Orient the undirected edges among $\{X_i, X_j, X_k\}$ independently according to $\text{Bern}(\frac{1}{2})$ until it becomes a directed structure which is not a directed cycle or a $v$-structure which was not in $\mathcal{Ess}(G^*)$.

end for

end while

$G^i = G^i \cup G^i$

end for

Output: $\hat{D}(\mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{G^i \in \mathcal{G}^i} D(\mathcal{I}, G^i)$

---

**Proposition 1** (Chernoff Bound). Let $X_1, \ldots, X_N$ be independent random variables such that for all $i$, $0 \leq X_i \leq 1$. Let $\mu = E[\sum_{i=1}^{N} X_i]$. Then

$$P\left(\sum_{i=1}^{N} X_i - \mu \geq \epsilon \mu\right) \leq 2 \exp\left(-\frac{\epsilon^2}{2+\epsilon} \mu\right).$$

**Theorem 3.** For an estimator with $E[D(\mathcal{I}, G^i)] = \mathcal{D}(\mathcal{I})$, given set $\mathcal{I}$ and $\epsilon, \delta > 0$, if $N = \left|\mathcal{G}^i\right| > \frac{|E(\mathcal{Ess}(G^*))|(2+\epsilon)}{\epsilon^2}$, then

$$\mathcal{D}(\mathcal{I})(1 - \epsilon) < \hat{\mathcal{D}}(\mathcal{I}) < \mathcal{D}(\mathcal{I})(1 + \epsilon),$$

with probability larger than $1 - \delta$.

**Proof.** Define $X_i = \frac{D(\mathcal{I}, G^i)}{|E(\mathcal{Ess}(G^*))|}$, for $i \in \{1, \ldots, N\}$.

By the assumption of the theorem, $E[X_i] = \frac{|E(\mathcal{Ess}(G^*))|}{|E(\mathcal{Ess}(G^*))|} \mathcal{D}(\mathcal{I})$

(Note that we saw in the proof of Theorem 2 that this assumption is satisfied). Using Chernoff bound we have

$$P\left(\sum_{i=1}^{N} X_i - \mu \geq \epsilon \mu\right) = P\left(\sum_{i=1}^{N} X_i - \mathcal{D}(\mathcal{I}) \geq \epsilon \mathcal{D}(\mathcal{I})\right) \leq 2 \exp\left(-\frac{\epsilon^2}{2+\epsilon} \mathcal{D}(\mathcal{I})\right).$$

Therefore,

$$P\left(\sum_{i=1}^{N} X_i - \mathcal{D}(\mathcal{I}) \geq \epsilon \mathcal{D}(\mathcal{I})\right) \leq 2 \exp\left(-\frac{\epsilon^2}{|E(\mathcal{Ess}(G^*))|} \mathcal{D}(\mathcal{I})\right).$$

Setting $N > \frac{|E(\mathcal{Ess}(G^*))|(2+\epsilon)}{\epsilon^2}$, upper bounds the right hand side with $1 - \delta$ and concludes the desired result. \qed

---

**Lemma 3.** Each generated directed graph $G^i$ in $D(\mathcal{I})$ Estimator Subroutine belongs to the Markov equivalence class of $G^*$.

**Proof.** We require the following claim for the proof:

**Claim 1.** A chordal graph has a directed cycle only if it has a directed cycle of size 3.

To prove Claim 1, if the directed cycle is of size 3 itself, the claim is trivial. Suppose the cycle $C_n$ is of size $n > 3$. Relabel the vertices of $C_n$ to have $C_n = (v_1, \ldots, v_n, v_1)$. Since the graph is chordal, $C_n$ has a chord and hence we have a triangle on vertices $\{v_i, v_{i+1}, v_{i+2}\}$ for some $i$. If the direction of $\{v_i, v_{i+1}\}$ is $(v_{i+1}, v_i)$, we have the directed cycle $(v_i, v_{i+1}, v_{i+2}, v_{i+3})$, which is of size 3. Otherwise, we have the directed cycle $C_{n-1} = (v_1, \ldots, v_i, v_{i+2}, \ldots, v_n, v_1)$ on $n-1$ vertices. Relabeling the vertices from 1 to $n-1$ and repeating the above reasoning concludes the claim.

All the components in the undirected subgraph of $\mathcal{Ess}(G^*)$ are chordal (Hauser and Bühlmann 2012). Therefore, by Claim 1, to insure that a generated directed graph is a DAG, it suffices to make sure that it does not have any directed cycles of length 3, which is one of the checks that we do in the proposed procedure. For checking if the generated DAG is in the same Markov equivalence class as $G^*$, it suffices to check if they have the same set of $v$-structures (Verma and Pearl 1991), which is the other check that we do in the proposed procedure. \qed

---

A chord of a cycle is an edge not in the cycle whose endpoints are in the cycle. A hole in a graph is a cycle of length at least 4 having no chord. A graph is chordal if it has no hole.
Our general greedy algorithm is presented in Algorithm 1. Defining the marginal gain of variable $v$ when the previous chosen set is $\mathcal{I}$ as $\Delta_v(\mathcal{I}) = D(\mathcal{I} \cup \{v\}) - D(\mathcal{I})$, the algorithm iteratively adds a variable which has the largest marginal gain to the intervention target set until it runs out of budget. Either $D(\mathcal{I})$ estimators can be used in calculating an estimation of the marginal gains.

**Theorem 4.** For any $\epsilon', \delta' > 0$, there exists $\epsilon, \delta > 0$, such that if for any set $\mathcal{I}$, $D(\mathcal{I})(1 - \epsilon) < D(\mathcal{I}) < D(\mathcal{I})(1 + \epsilon)$ with probability larger than $1 - \delta$, then Algorithm 1 is a $(1 - \frac{1}{2} - \epsilon')$-approximation algorithm with probability larger than $1 - \delta'$.

See the appendix for the proof.

**Improved Algorithm**

Submodularity of function $D$ can be exploited to implement an accelerated variant of the general greedy algorithm through lazy evaluations, originally proposed by Minoux (Minoux 1978). In each round of the general greedy algorithm, we check the marginal gain $\Delta_v(\mathcal{I})$ for all remaining vertices in $\mathcal{V} \setminus \mathcal{I}$. Note that as a consequence of submodularity of function $D$, the set function $\Delta_v$ is monotonically decreasing. The main idea of the improved greedy algorithm is to take advantage of this property to avoid checking all the variables in each round of the algorithm. The idea is as follows: Suppose for vertices $v_1$ and $v_2$, in the $i$-th round of the algorithm we have obtained marginal gains $\Delta_{v_1}(\mathcal{I}_i) > \Delta_{v_2}(\mathcal{I}_i)$. Now, if in the $(i + 1)$-th round we have calculate $\Delta_{v_1}(\mathcal{I}_{i+1})$ and have $\Delta_{v_1}(\mathcal{I}_{i+1}) > \Delta_{v_2}(\mathcal{I}_i)$, then by monotonic decreasing property, we can conclude that $\Delta_{v_1}(\mathcal{I}_{i+1}) > \Delta_{v_2}(\mathcal{I}_{i+1})$, and hence, there is no need to calculate $\Delta_{v_2}(\mathcal{I}_{i+1})$.

The detailed description of the improved greedy algorithm is given in Algorithm 2. The idea can be formalized as follows (Leskovec et al. 2007): We define a profit parameter $p_v$ for each variable $v$ and initialize the value of all of them with $\infty$. Also, for all variables we define an update flag $upd_v$, which will be set to false at the beginning of every round of the algorithm and will be switched to true if we update $p_v$ with the value of the marginal gain of vertex $v$. In each round of the algorithm, we pick vertex $v \in \mathcal{V} \setminus \mathcal{I}$ with the largest profit, update its profit with the value of the marginal gain of $v$, and set $upd_v$ to true. This process is repeated until the vertex with the largest profit is already updated, i.e., its update flag is true. Then we add this vertex to $\mathcal{I}$ and end the round. For example, if in a round, the vertex $v$ has the highest profit and after updating the profit of this vertex, $p_v$ is still larger than all the other profits, we do not need to evaluate the marginal gain of any other vertex and we add $v$ to $\mathcal{I}$.

The correctness of the improved algorithm follows directly from submodularity and Theorem 4 holds for Algorithm 2 as well. This algorithm can lead to orders of magnitude performance speedup (Leskovec et al. 2007).

**Synthetic Graphs**

In this subsection, we evaluate the performance of the proposed improved greedy algorithm on synthetically generated chordal graphs. We use randomly chosen partial elimination ordering (PEO) of the vertices to generate our underlying chordal graphs (Hauser and Buhlmann 2014; Shanmugam et al. 2015). For each graph, we pick a random ordering of the vertices. Starting from the vertex $v$ with the highest order, we connect all the vertices with lower order to $v$ with probability inversely proportional to the order of $v$. Then, we connect all the parents of $v$ with directed edges, where each directed edge is oriented from the parent with the lower order to the parent with the higher order. In order to make sure that the generated graph will be connected, if vertex $v$ is not connected to any of the vertices with the lower order, we pick one of them uniformly at random and set it as the parent of $v$.

To evaluate the performance of the proposed algorithm, for any underlying graph, we consider the ratio of the number of edges whose directions are discovered merely as a result of interventions to the number of edges whose directions were not resolved from the observational data. Note that due to our specific graph generating approach, the orientation of none of the edges is learned from the observational data. We

---

**Algorithm 1** General Greedy Algorithm

**Input:** Joint distribution over $\mathcal{V}$, and budget $k$.

Obtain $E_{ss}(G^*)$ by performing a CCI algorithm.

**Initial:** $\mathcal{I}_0 = \emptyset$

for $i = 1$ to $k$

$\nu_i = \arg \max_{v \in \mathcal{V} \setminus \mathcal{I}_{i-1}} D(\mathcal{I}_{i-1} \cup \{v\}) - D(\mathcal{I}_{i-1})$

$\mathcal{I}_i = \mathcal{I}_{i-1} \cup \{\nu_i\}$

end for

**Output:** $\hat{\mathcal{I}} = \mathcal{I}_k$

**Algorithm 2** Improved Greedy Algorithm

**Input:** Joint distribution over $\mathcal{V}$, and budget $k$.

Obtain $E_{ss}(G^*)$ by performing a CCI algorithm.

**Initial:** $\mathcal{I}_0 = \emptyset$, and $p_v = \infty$, $\forall v \in \mathcal{V}$.

for $i = 1$ to $k$

$\nu^* = \arg \max_{v \in \mathcal{V} \setminus \mathcal{I}_{i-1}} p_v$

if $\nu^*$, then

$\mathcal{I}_i = \mathcal{I}_{i-1} \cup \{\nu^*\}$

break;

else

$p_{\nu^*} = D(\mathcal{I}_{i-1} \cup \{\nu^*\}) - D(\mathcal{I}_{i-1})$

$upd_{\nu^*} = \text{true}$

end if

end while

**Output:** $\hat{\mathcal{I}} = \mathcal{I}_k$

---

A perfect elimination ordering $\{v_1, v_2, ..., v_n\}$ on the vertices of an undirected chordal graph is such that for all $i$, the induced neighborhood of $v_i$ on the subgraph formed by $\{v_1, v_2, ..., v_i\}$ is a clique.
generated 100 instances of chordal DAGs of order 20. In all experiments, the improved greedy algorithm is utilized and Subroutine 2 is used for estimating $D(I)$. Figure 1 depicts the discovered edge ratio with respect to the budget $k$. As seen in this figure, three interventions suffice to discover the direction of more than 90% of the edges whose direction was unknown prior to performing interventions. Further to investigate the effect of the order of the graph on the performance of the proposed algorithm, we evaluated the discovered edge ratio for budget $k = 3$ on graphs with order $n \in \{10, 15, 20, 25, 30\}$ in Figure 2.

Furthermore, to compare the performance of the proposed algorithm with the optimal solution, we generated 100 instances of chordal DAGs of order 10 and performed a brute force search to find the optimal solution of $\{1\}$ for budget $k = 2$. The discovered edge ratio was 0.9 and 0.916 for our proposed algorithm and the optimal solution, respectively.

**Real Graphs**

We evaluated the performance of the proposed algorithm in gene regulatory networks (GRN). GRN is a collection of biological regulators that interact with each other. In GRN, the transcription factors are the main players to activate genes. The interactions between transcription factors and regulated genes in a species genome can be presented by a directed graph. In this graph, links are drawn whenever a transcription factor regulates a gene’s expression. Moreover, some of vertices have both functions, i.e., are both transcription factor and regulated gene.

We considered GRNs in “DREAM 3 In Silico Network” challenge, conducted in 2008 (Marbach et al. 2009). The networks in this challenge were extracted from known biological interaction networks. Since we know the true causal structures in these GRNs, we can obtain $\text{Ess}(G^*)$ and give it as an input to the proposed algorithm. Figure 3 depicts the discovered edge ratio in five networks extracted from GRNs of E-coli and Yeast bacteria with budget $k = 5$. The order of each network is 100. As it can be seen, the discovered edge ratio is at least 0.65 in all GRNs.

**Conclusion**

We studied the problem of experiment design for causal inference when only a limited number of experiments are available. In our model, each experiment consists of intervening on a single vertex, which makes results in taking the most advantage from a finite budget of interventions and makes the model suitable for applications in which intervening on several variables simultaneously is not feasible. Also, in our model, experiments are designed merely based on the result of an initial purely observational test, which enables the experimenter to perform the interventional tests in parallel. We addressed the following question: “How much of the causal structure can be learned when only a limited number of experiments are available?” We formulated the problem of finding the best intervention target set as an optimization problem which aims to maximize the average number of edges whose directions are discovered. We showed that the objective function is monotonically increasing and submodular. Consequently, the greedy algorithm is a $(1 - \frac{1}{e})$-approximation algorithm for this problem. Moreover, we proposed estimation methods in order to compute the objective function for a given set of intervention targets. We further presented an accelerated variant of the greedy algorithm which can lead to orders of magnitude performance speedup. We examined our proposed improved greedy algorithm on synthetic as well as real graphs. The results show that a significant portion of causal graphs can be learned by only a few number of interventions.
Appendices

Example of Comparison with the Influence Maximization Problem

Suppose k = 1. Figure 3 depicts a graph for which the optimal solution to the influence maximization problem is different from the optimal solution to the budgeted experiment design problems. Clearly, influencing vertex v1 leads to influencing all the vertices in the graph, and hence, this vertex is the solution to the influence maximization problem. But, intervening on v1 leads to discovering the orientation of only 3 edges, while intervening on, say v2, leads to discovering the orientation of 5 edges.

Proof of Lemma 2

The direction \( R(A_{G^*}^{(I)}) ) \subseteq \bigcup_{(a,b)} R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \) is proved in the proof of Lemma 1.

Also, as observed in the proof of Lemma 1, we have \( R(A_{G^*}^{(I_a)}, G^*) \subseteq R(R(A_{G^*}^{(I_a)}), G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \).

Therefore, in order to prove that \( R(A_{G^*}^{(I_a)}, G^*) \subseteq R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \), it suffices to show that \( R(R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \subseteq R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \), for which it suffices to show that if \( e \notin R(A_{G^*}^{(I_a)}, G^*) \) and \( e \notin R(A_{G^*}^{(I_b)}, G^*) \), then \( e \notin R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \).

Proof by contradiction. Let \( e \notin R(A_{G^*}^{(I_a)}, G^*) \) and \( e \notin R(A_{G^*}^{(I_b)}, G^*) \), but its orientation is learned in the first iteration of applying Meek rules to \( R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \cup A(Ess(G^*)). \) Then, we have learned the orientation of \( e \) due to one of Meek rules Verma and Pearl (1992):

- **Rule 1.** \( e = \{a,b\} \) is oriented as \( (a,b) \) if \( \exists c \) s.t. \( e_1 = (c,a) \in R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup A(Ess(G^*)) \), and \( \{c,b\} \notin \text{skeleton of } G^* \).

- **Rule 2.** \( e = \{a,b\} \) is oriented as \( (a,b) \) if \( \exists c \) s.t. \( e_1 = (a,c) \in R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup A(Ess(G^*)) \), and \( e_2 = (c,b) \in R(A_{G^*}^{(I_c)}, G^*) \cup R(A_{G^*}^{(I_d)}, G^*) \cup A(Ess(G^*)).

- **Rule 3.** \( e = \{a,b\} \) is oriented as \( (a,b) \) if \( \exists c, d \) s.t. \( e_1 = (c,b) \in R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_b)}, G^*) \cup A(Ess(G^*)) \), and \( e_2 = (d,b) \in R(A_{G^*}^{(I_c)}, G^*) \cup R(A_{G^*}^{(I_d)}, G^*) \cup A(Ess(G^*)) \), \( \{a,c\} \in \text{skeleton of } G^* \), \( \{a,d\} \in \text{skeleton of } G^* \), and \( \{c,d\} \notin \text{skeleton of } G^* \).

- **Rule 4.** \( e = \{a,b\} \) is oriented as \( (a,b) \) and \( e = \{b,c\} \) is oriented as \( (c,b) \) if \( \exists d \) s.t. \( e_1 = (d,c) \in R(A_{G^*}^{(I_a)}, G^*) \cup R(A_{G^*}^{(I_c)}, G^*) \cup A(Ess(G^*)) \), \( \{a,c\} \in \text{skeleton of } G^* \), \( \{a,d\} \in \text{skeleton of } G^* \), and \( \{b,d\} \notin \text{skeleton of } G^* \).

In what follows, we show that the orientation of \( e \) cannot be learned due to any of the Meek rules unless it belongs to

![Figure 4: Example of comparison with the influence maximization problem](image)

\( R(A_{G^*}^{(I_1)}), G^* ) \) or \( R(A_{G^*}^{(I_2)}), G^* ). \)

**Rule 1.**

Without loss of generality, assume \( e_1 \in R(A_{G^*}^{(I_1)}), G^* ) \) or \( A(Ess(G^*)) \). Therefore, we should have the condition of rule 1 satisfied when only intervening on \( I_1 \) as well, which implies that \( e \in R(A_{G^*}^{(I_1)}), G^* ) \), which is a contradiction.

**Rule 2.**

If both \( e_1 \) and \( e_2 \) belong to \( R(A_{G^*}^{(I_1)}), G^* ) \) or \( A(Ess(G^*)) \) or \( R(A_{G^*}^{(I_2)}), G^* ) \) or \( A(Ess(G^*)) \) or \( R(A_{G^*}^{(I_3)}), G^* ) \) or \( A(Ess(G^*)) \) and \( e_2 \) belongs only to the other one, does not happen. To this end, it suffices to show that there does not exist intervention target set \( \mathcal{T} \) such that \( e_1 \in R(A_{G^*}^{(I_1)}), G^* ) \) or \( A(Ess(G^*)) \) and \( e_2 \notin R(A_{G^*}^{(I_2)}), G^* ) \) or \( A(Ess(G^*)) \), i.e., there does not exist intervention target set \( \mathcal{T} \) that has structure \( S_0 \), depicted in Figure 5 as a subgraph of \( Ess(G^*) \) after applying the orientations learned from \( R(A_{G^*}^{(I_1)}), G^* ). \)

![Figure 5: Structure S0](image)

If \( e_1 \in A_{G^*}^{(I_1)} \), then \( a \in \mathcal{T} \) or \( c \in \mathcal{T} \), which implies \( e \notin A_{G^*}^{(I_1)} \) or \( e \notin A_{G^*}^{(I_2)} \), respectively, and hence, \( e \notin R(A_{G^*}^{(I_1)}), G^* ) \) or \( e \notin R(A_{G^*}^{(I_2)}), G^* ) \), respectively. Therefore, in either case, \( e \notin R(A_{G^*}^{(I_1)}), G^* ) \), and \( S_0 \) will not be a subgraph. Therefore, \( e_1 \notin A_{G^*}^{(I_1)} \), and hence, \( e_1 \) was learned by applying one of the Meek rules. We consider each or the rules in the following:

- If we have learned the orientation of \( e_1 \) from rule 1, then we should have had one of the structures in Figure 5 as a subgraph of \( Ess(G^*) \) after applying the orientations learned from \( R(A_{G^*}^{(I_1)}), G^* ). \) In case of structure \( S_1 \), using rule 1 on subgraph induced on vertices \( \{v_1, a, b\} \), we will also learn \( \{a, b\} \). In case of structure \( S_2 \), using rule 4, we will also learn \( \{b, c\} \). Therefore, we cannot learn only the direction of \( e_1 \) and hence, \( S_0 \) will not be a subgraph.
If we have learned the orientation of $e_1$ from rule 3, then we have had one of the structures in Figure 7 as a subgraph of $\text{Ess}(G^*)$ after applying the orientations learned from $R(\tilde{A}^{(E)}_G, G^*)$. In case of structures $S_3$ and $S_4$, using rule 1 on subgraph induced on vertices $\{v_2, c, b\}$, we will also learn $(c, b)$. In case of structure $S_5$, using rule 3 on subgraph induced on vertices $\{b, v_2, c, v_1\}$, we will also learn $(b, c)$. Therefore, we cannot learn only the direction of $e_1$ and hence, $S_0$ will not be a subgraph.

If we have learned the orientation of $e_1$ from rule 4, then we have had one of the structures in Figure 8 as a subgraph of $\text{Ess}(G^*)$ after applying the orientations learned from $R(\tilde{A}^{(E)}_G, G^*)$. In case of structures $S_6$, using rule 1 on subgraph induced on vertices $\{v_1, c, b\}$, we will also learn $(c, b)$. In case of structure $S_7$, using rule 1 on subgraph induced on vertices $\{v_2, v_1, b\}$, we will also learn $(v_1, b)$, and then using rule 4 on subgraph induced on vertices $\{b, a, v_2, v_1\}$, we will also learn $(a, b)$. In case of structures $S_8$, using rule 4 on subgraph induced on vertices $\{b, v_2, v_1, c\}$, we will also learn $(b, c)$. Therefore, we cannot learn only the direction of $e_1$ and hence, $S_0$ will not be a subgraph.

If we have learned the orientation of $e_1$ from rule 2, then we should have had one of the structures in Figure 9 as a subgraph of $\text{Ess}(G^*)$ after applying the orientations learned from $R(\tilde{A}^{(E)}_G, G^*)$. In case of structure $S_9$, using rule 1 on subgraph induced on vertices $\{v_2, c, b\}$, we will also learn $(c, b)$ and hence, $S_0$ will not be a subgraph. In case of structure $S_{10}$, if $v_1 \in I$, then the direction of the edge $\{v_1, b\}$ will be also known. If the direction of this edge is $(v_1, b)$, then using rule 2 on subgraph induced on vertices $\{a, v_1, b\}$, we will also learn $(a, b)$; otherwise, using rule 2 on subgraph induced on vertices $\{b, v_1, c\}$, we will also learn $(c, b)$. Therefore, $v_1 \notin I$. Also, as mentioned earlier, $a \notin I$. Therefore, we have learned the orientation of $u_1$ from applying Meek rules.

In the triangle induced on vertices $\{v_1, b, a\}$, we have learned only the orientation of one edge, which is $(a, v_1)$. But as seen in structures $S_1$ to $S_8$, all of them lead to learning the orientation of at least 2 edges of a triangle. In the following, we will show that a structure of form $S_{10}$ does not lead to learning the orientation of only $(a, v_1)$ and making $S_{10}$ a subgraph either.

We claim that this procedure always repeats, i.e., at step $i$, we end up with skeleton $K_{i-1}$, which contains $\{v_j, c\}, \{a, v_j\}, \{v_j, b\}, \{v_k, v_j\}$, for $j \in \{1, \ldots, i\}$ and $(v_k, v_j)$, for $1 \leq j < k \leq i$, with a triangle induced on vertices $\{v_i, b, a\}$, in which we have learned only the orientation of $(a, v_i)$. We prove this claim by induction. We have already proved the base of the induction above. For the step of the induction, suppose the hypothesis is true for $i - 1$. Add vertex $v_i$ to form a structure of form $S_{11}$ for $(a, v_{i-1})$. $v_i$ should be adjacent to $v_j$, for $j \in \{1, \ldots, i - 2\}$; otherwise, using rule 4 on subgraph induced on vertices $\{v_i, v_{i-1}, v_j, b\}$, we will also learn $(b, v_j)$. Moreover, using rule 2 on triangle induced on vertices $\{v_i, v_{i-1}, v_j\}$, the direction of
\{v_i, v_j\} should be \(\{v_i, v_j\}\). Also, using rule 4 on subgraph induced on vertices \(\{v_i, v_{i-1}, c, b\}\), we will also learn \((b, c)\). Therefore, we should have the edge \(\{v_i, c\}\)
too.

We showed that \(S_0\) is a subgraph only if \(S_10\) is a subgraph, and \(S_10\) is a subgraph only if the structure in Figure 10(b) is a subgraph, and this chain of required subgraphs continue. Therefore, since the order of the graph is finite, there exist a step where since we cannot add a new vertex, it is not possible to have one of the required subgraphs, and hence we conclude that \(S_0\) is not a subgraph.

**Rule 3.**

Since edges \(e_1\) and \(e_2\) form a v-structure, they should appear in \(A(E_{\text{Ess}}(G^*))\) as well. Therefore, we should have the condition of rule 3 satisfied when only intervening on \(\mathcal{I}_1\) as well, which implies that \(e \in R(A(G^*_1), G^*)\), which is a contradiction.

**Rule 4.**

Without loss of generality, assume \(e_1 \in R(A(G^*_1), G^*) \cup A(E_{\text{Ess}}(G^*))\). Therefore, we should have the condition of rule 4 satisfied when only intervening on \(\mathcal{I}_1\) as well, which implies that \(e \in R(A(G^*_1), G^*)\), which is a contradiction.

The argument above proves that there is no edge \(e\) such that \(e \notin R(A(G^*_1), G^*)\) and \(e \notin R(A(G^*_1), G^*)\), but \(e \in R(R(A(G^*_1), G^*) \cup R(A(G^*_1), G^*), G^*)\).

**Proof of Theorem 2**

Let \(G_{\text{total}}\) be all directed graphs with the same skeleton as \(G^*\) and \(G' \in G_{\text{total}}\).

\[
E[D(\mathcal{I}, G')] = \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)
\]

\[
= \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)
+ \sum_{G' \notin G_{\text{total}}} D(\mathcal{I}, G') P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)
\]

\[
= \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)
+ \sum_{G' \notin G_{\text{total}}} \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') \times 0
\]

\[
= \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') \frac{P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)}{P(G' \in G)}
\]

\[
= \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') \frac{P(\text{outputing } \hat{G}' = G' | \hat{G}' \in G)}{P(G' \in G)}
\]

Since we can orient \(E(E_{\text{Ess}}(G^*))\) in \(2^{E(E_{\text{Ess}}(G^*))}\) ways and noting that we used the notation \(G^*\) for those DAGs which are in the Markov equivalence of \(G^*\), we have

\[
E[D(\mathcal{I}, G')] = \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') \frac{1}{2^{E(E_{\text{Ess}}(G^*))}}
\]

\[
= \frac{1}{|G|} \sum_{G' \in G_{\text{total}}} D(\mathcal{I}, G') = D(\mathcal{I}).
\]

Therefore, with \(N := |G'|\), we have

\[
E[D(\mathcal{I})] = E_{\mathcal{I}} \frac{1}{N} \sum_{i=1}^{N} D(\mathcal{I}, G_i') = E_{\mathcal{I}} \left[ \frac{1}{N} \sum_{i=1}^{N} D(\mathcal{I}, G_i') \right] |N|
\]

\[
= E_{\mathcal{I}} \left[ \frac{1}{N} \sum_{i=1}^{N} E[D(\mathcal{I}, G_i')] \right] = E_{\mathcal{I}} \left[ \frac{1}{N} \sum_{i=1}^{N} E[D(\mathcal{I}, G_i')] \right]
\]

\[
= E_{\mathcal{I}} \left[ \frac{1}{N} \sum_{i=1}^{N} D(\mathcal{I}) \right] = E_{\mathcal{I}} \left[ \frac{1}{N} \sum_{i=1}^{N} D(\mathcal{I}) \right] = D(\mathcal{I}).
\]

**Proof of Theorem 4**

Let \(\mathcal{I}^* = \{v^*_1, ..., v^*_k\} \in \arg\max_{\mathcal{I} \subseteq V, |\mathcal{I}| = k} D(\mathcal{I})\). We have

\[
D(\mathcal{I}^*) \leq D(\mathcal{I}^* \cup \mathcal{I}_1) = D(\mathcal{I}_1)
\]

\[
+ \sum_{j=1}^{k} [D(\mathcal{I}_1 \cup \{v^*_1, ..., v^*_j\}) - D(\mathcal{I}_1 \cup \{v^*_1, ..., v^*_j-1\})]
\]

\[
\leq D(\mathcal{I}_1) + \sum_{j=1}^{k} [D(\mathcal{I}_1 \cup \{v^*_j\}) - D(\mathcal{I}_1)],
\]

where (a) follows from Lemma 1 and (b) follows from Theorem 1. Define \(\hat{D}_{i,v^*_1,1}(\mathcal{I}_i)\) and \(\hat{D}_{i,v^*_1,2}(\mathcal{I}_i)\) as the first and second calls of subroutine in \(i\)-th step for variable \(v^*_i\), respectively. By the assumption of the theorem we have

\[
D(\mathcal{I}_1 \cup \{v^*_j\}) < cD(\mathcal{I}_1 \cup \{v^*_j\}) < \hat{D}_{i,v^*_1,1}(\mathcal{I}_i \cup \{v^*_j\}),
\]

with probability larger than \(1 - \delta\). Therefore

\[
D(\mathcal{I}_1 \cup \{v^*_j\}) < \hat{D}_{i,v^*_1,1}(\mathcal{I}_i \cup \{v^*_j\}) + cD(\mathcal{I}^*),
\]

with probability larger than \(1 - \delta\). Similarly

\[
\hat{D}_{i,v^*_2,1}(\mathcal{I}_i) < D(\mathcal{I}_i) + cD(\mathcal{I}_i) \quad w.p. > 1 - \delta,
\]

\[
\Rightarrow \hat{D}_{i,v^*_2,1}(\mathcal{I}_i) < \hat{D}_{i,v^*_2,1}(\mathcal{I}_i) + cD(\mathcal{I}^*) \quad w.p. > 1 - \delta.
\]

Therefore,

\[
D(\mathcal{I}_1 \cup \{v^*_j\}) - D(\mathcal{I}_1) < \hat{D}_{i,v^*_1,1}(\mathcal{I}_i \cup \{v^*_j\}) - \hat{D}_{i,v^*_2,1}(\mathcal{I}_i) + 2cD(\mathcal{I}^*) \quad w.p. > 1 - 2\delta.
\]

Also, by the definition of the greedy algorithm,

\[
\hat{D}_{i,v^*_1,1}(\mathcal{I}_i \cup \{v^*_j\}) - \hat{D}_{i,v^*_2,1}(\mathcal{I}_i) \leq \hat{D}_{i,v^*_1,1}(\mathcal{I}_i \cup \{v^*_j\}) - \hat{D}_{i,v^*_1,2}(\mathcal{I}_i) \quad (4)
\]

\[
= \hat{D}_{i,v^*_1,1}(\mathcal{I}_i) - \hat{D}_{i,v^*_1,2}(\mathcal{I}_i),
\]
and similar to (3), we have
\[
D_{i,v_i+1}(I_{i+1}) - D_{i,v_i+1,2}(I_i) < D(I_{i+1}) - D(I_i) + 2\epsilon D(I^*) \quad w.p. > 1 - 2\delta.
\]
Therefore, from equations (3), (4), and (5) we have
\[
D(I_i \cup \{v_i^+\}) - D(I_i) < D(I_{i+1}) - D(I_i) + 4\epsilon D(I^*),
\]
with probability larger than 1 – 4δ. Plugging (6) back in (2), we get
\[
D(I^*) < D(I_i) + \sum_{j=1}^{k} D(I_{i+1}) - D(I_i) + 4\epsilon D(I^*) = D(I_i) + k[D(I_{i+1}) - D(I_i)] + 4\epsilon k D(I^*),
\]
with probability larger than 1 – 4kδ. Therefore,
\[
D(I^*) - D(I_i) < k[D(I^*) - D(I_i)] - k[D(I^*) - D(I_{i+1})] + 4\epsilon k D(I^*),
\]
with probability larger than 1 – 4kδ. Defining \(a_k := D(I^*) - D(I_i)\), and noting that \(a_0 = D(I^*)\), by induction we have
\[
a_k = D(I^*) - D(I_k) < (1 - \frac{1}{k})^k D(I^*) + 4\epsilon k D(I^*) \sum_{j=0}^{k-1} (1 - \frac{1}{k})^j < \frac{1}{e} + 4\epsilon k D(I^*) \quad w.p. > 1 - 4k^2 \delta.
\]
It concludes that
\[
D(I_k) > (1 - \frac{1}{e} - 4\epsilon k) D(I^*) \quad w.p. > 1 - 4k^2 \delta.
\]
Therefore, for \(\epsilon = \frac{\epsilon'}{4k} \) and \(\delta = \frac{\epsilon'}{4k^2\pi}\), Algorithm [1] is a \((1 - \frac{1}{e} - \epsilon')\)-approximation algorithm with probability larger than 1 – 4\(\epsilon\).

References

[Chen, Wang, and Yang 2009] Chen, W.; Wang, Y.; and Yang, S. 2009. Efficient influence maximization in social networks. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, 199–208. ACM.

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

[Cooper and Yoo 1999] Cooper, G. F., and Yoo, C. 1999. Causal discovery from a mixture of experimental and observational data. In Proceeding of the Fifteenth conference on Uncertainty in artificial intelligence, 116–125. Morgan Kaufmann Publishers Inc.

[Eberhardt, Glymour, and Scheines 2005] Eberhardt, F.; Glymour, C.; and Scheines, R. 2005. On the number of experiments sufficient and in the worst case necessary to identify all causal relations among n variables. 178–184.

[Eberhardt, Glymour, and Scheines 2006] Eberhardt, F.; Glymour, C.; and Scheines, R. 2006. N-1 experiments suffice to determine the causal relations among n variables. In Innovations in machine learning. Springer. 97–112.

[Eberhardt 2007] Eberhardt, F. 2007. Causation and intervention. Unpublished doctoral dissertation, Carnegie Mellon University.

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

[Hauser and Bühlmann 2012] Hauser, A., and Bühlmann, P. 2012. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. Journal of Machine Learning Research 13(Aug):2409–2464.

[Hauser and Bühlmann 2014] Hauser, A., and Bühlmann, P. 2014. Two optimal strategies for active learning of causal models from interventional data. International Journal of Approximate Reasoning 55(4):926–939.

[He and Geng 2008] He, Y.-B., and Geng, Z. 2008. Active learning of causal networks with intervention experiments and optimal designs. Journal of Machine Learning Research 9(Nov):2523–2547.

[Hoyer et al. 2009] Hoyer, P. O.; Janzing, D.; Mooij, J. M.; Peters, J.; and Schölkopf, B. 2009. Nonlinear causal discovery with additive noise models. In Advances in neural information processing systems, 689–696.

[Hyttinen, Eberhardt, and Hoyer 2013] Hyttinen, A.; Eberhardt, F.; and Hoyer, P. O. 2013. Experiment selection for causal discovery. Journal of Machine Learning Research 14(1):3041–3071.

[Kempe, Kleinberg, and Tardos 2003] Kempe, D.; Kleinberg, J.; and Tardos, É. 2003. Maximizing the spread of influence through a social network. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, 137–146. ACM.

[Kocaoglu, Dimakis, and Vishwanath 2017] Kocaoglu, M.; Dimakis, A. G.; and Vishwanath, S. 2017. Cost-optimal learning of causal graphs. arXiv preprint arXiv:1703.02645.

[Koller and Friedman 2009] Koller, D., and Friedman, N. 2009. Probabilistic graphical models: principles and techniques. MIT press.

[Leskovec et al. 2007] Leskovec, J.; Krause, A.; Guestrin, C.; Faloutsos, C.; VanBriesen, J.; and Glance, N. 2007. Cost-effective outbreak detection in networks. In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, 420–429. ACM.

[Marbach et al. 2009] Marbach, D.; Schaffer, T.; Mattiussi, C.; and Floreano, D. 2009. Generating realistic in silico gene networks for performance assessment of reverse engineering methods. Journal of computational biology 16(2):229–239.

[Meek 1997] Meek, C. 1997. Graphical models: Selecting causal and statistical models.

[Meinshausen et al. 2016] Meinshausen, N.; Hauser, A.; Mooij, J. M.; Peters, J.; Versteeg, P.; and Bühlmann, P. 2016. Methods for causal inference from gene perturbation experiments and validation. Proceedings of the National Academy of Sciences 113(27):7361–7368.

[Minoux 1978] Minoux, M. 1978. Accelerated greedy algo-
rithms for maximizing submodular set functions. *Optimization Techniques* 234–243.

[Nemhauser, Wolsey, and Fisher 1978] Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functions?i. *Mathematical Programming* 14(1):265–294.

[Pearl and Bareinboim 2011] Pearl, J., and Bareinboim, E. 2011. External validity and transportability: A formal approach. *JSM Proceedings, Miami Beach FL* 157–171.

[Pearl 2009] Pearl, J. 2009. *Causality*. Cambridge university press.

[Peters and Bühlmann 2012] Peters, J., and Bühlmann, P. 2012. Identifiability of gaussian structural equation models with equal error variances. *arXiv preprint arXiv:1205.2536*.

[Peters, Bühlmann, and Meinshausen 2016] Peters, J.; Bühlmann, P.; and Meinshausen, N. 2016. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 78(5):947–1012.

[Schölkopf et al. 2012] Schölkopf, B.; Janzing, D.; Peters, J.; Sgouritsa, E.; Zhang, K.; and Mooij, J. 2012. On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*.

[Shanmugam et al. 2015] Shanmugam, K.; Kocaoglu, M.; Dimakis, A. G.; and Vishwanath, S. 2015. Learning causal graphs with small interventions. In *Advances in Neural Information Processing Systems*, 3195–3203.

[Shimizu et al. 2006] 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(Oct):2003–2030.

[Shpitser and Pearl 2006] Shpitser, I., and Pearl, J. 2006. Identification of joint interventional distributions in recursive semi-markovian causal models. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, 1219.

[Shpitser, VanderWeele, and Robins 2012] Shpitser, I.; VanderWeele, T.; and Robins, J. M. 2012. On the validity of covariate adjustment for estimating causal effects. *arXiv preprint arXiv:1203.3515*.

[Spirtes, Glymour, and Scheines 2000] Spirtes, P.; Glymour, C. N.; and Scheines, R. 2000. *Causation, prediction, and search*. MIT press.

[Verma and Pearl 1991] Verma, and Pearl, J. 1991. Equivalence and synthesis of causal models. In *Proceedings of UAI*, 220–227.

[Verma and Pearl 1992] Verma, T., and Pearl, J. 1992. An algorithm for deciding if a set of observed independencies has a causal explanation. In *Proceedings of UAI*, 323–330.