Minimum Cost Intervention Design for Causal Effect Identification

Sina Akbari [1] **Jalal Etesami** [1] **Negar Kiyavash** [1]

Abstract

Pearl's *do calculus* is a complete axiomatic approach to learn the *identifiable* causal effects from observational data. When such an effect is not identifiable, it is necessary to perform a collection of often costly interventions in the system to learn the causal effect. In this work, we consider the problem of designing the collection of interventions with the minimum cost to identify the desired effect. First, we prove that this problem is NP-hard and subsequently propose an algorithm that can either find the optimal solution or a logarithmic-factor approximation of it. This is done by establishing a connection between our problem and the minimum hitting set problem. Additionally, we propose several polynomial time heuristic algorithms to tackle the computational complexity of the problem. Although these algorithms could potentially stumble on sub-optimal solutions, our simulations show that they achieve small regrets on random graphs.

1. Introduction

Causal inference plays a key role in many applications such as psychology (Foster, 2010), econometrics (Hoover, 1990), education, social sciences (Murnane & Willett, 2010; Gangl, 2010), etc. Causal effect identification, one of the most fundamental topics in causal inference, is concerned with estimating the effect of intervening on a set of variables, say X on another set of variables, say Y denoted by $P(Y|do(X))$. The estimation is performed having access to a set of observational and/or interventional distributions under causal assumptions that are usually encoded in the form of a causal graph. The causal graph of a system of variables captures the interconnection among the variables and can be inferred from a combination of observations, experiments, and expert knowledge about the phenomenon under investigation (Spirtes et al., 2000). Throughout this work, we assume that the causal graph is given as a side information.

Given a causal graph, it is known that in the absence of unobserved (latent) variables, every causal effect is identifiable from mere observational data (Robins, 1987; Spirtes et al., 2000). On the other hand, inferring causal effects from data becomes challenging in the presence of latent variables. In the setting where only observational data is available, the do-calculus, introduced by Pearl (Pearl, 1995), has been shown to be complete. That is, it provides a complete set of rules to compute a causal effect (if identifiable) given a causal graph and observational data (Huang & Valtorta, 2006). Moreover, polynomial time algorithms exist that can determine the identifiability of a causal effect using the do-calculus (Shpitser & Pearl, 2006).

In recent years, there has been an increase in the effort to generalize Pearl's do-calculus to the setting in which data from both observational and interventional data are available for identifying a causal effect. For instance, Bareinboim & Pearl (2012) studied the problem of estimating the causal effect of intervening on a set of variables X on the outcome Y when we experiment on a different set Z. This problem is known as z-identifiability, and (Bareinboim & Pearl, 2012) provides a complete algorithm for computing $P(Y|do(X))$ using information provided by experiments on all subsets of Z. A slightly more general version of z-identifiability is called g-identifiability, which considers the problem of identifying $P(Y|do(X))$ from an arbitrary collection of distributions. Lee et al. (2020) studied the g-identifiability problem and claimed that Pearl's do-calculus is also complete in this setting. This was proved by Kivva et al. (2022). All three of the aforementioned works study the identifiability of $P(Y|do(X))$.There are also various works that consider the more general problem of identifying a conditional causal effect of the form $P(Y|do(X), W)$. However, most of these works do not manage to provide complete results a la Pearl's do-calculus. See (Tikka et al., 2019), for a complete review on causal effect identification.

When a causal effect is not identifiable from the observations, it is necessary to perform a collection of interventions to infer the effect of interest. However, such interventions could be costly, impossible, or unethical to perform. Therefore, naturally we are interested in the problem of designing

[1]College of Management of Technology, EPFL. Correspondence to: Sina Akbari <sina.akbari@epfl.ch>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

a collection of low cost permitted interventions to identify a causal effect. This is the focus of our paper. A closely related work to ours is (Kandasamy et al., 2019), in which, the authors considered the problem of finding the minimum number of interventions to identify every possible causal query. Their approach is based on two limiting assumptions, namely that all interventions have the same cost and that we are allowed to intervene on any variable. More importantly, the result in (Kandasamy et al., 2019) guarantees to render any causal effect identifiable, which makes the solution sub-optimal. In other words, a set of interventions that makes all causal effects identifiable might have higher aggregate cost than a set of interventions designed for identifying a specific causal effect.

Designing minimum-cost interventions has also received attention in causal discovery literature, under the term *experimental design*. In causal discovery, the goal is to infer the causal graph from a dataset. It is known that mere observational data cannot fully recover the causal graph, and thus additional interventional data is required to precisely learn the graph. (Lindgren et al., 2018) considered the problem of designing a set with minimum number of interventions to learn a causal graph given the essential graph (assuming no latent variable), and showed that this problem is NP-hard. (Addanki et al., 2020) studied a similar problem in the presence of latent variables. The problem of orienting the maximum number of edges using a fixed number of interventions was studied in (Hauser & Bühlmann, 2014; Ghassami et al., 2018; Agrawal et al., 2019). (Addanki et al., 2021) studied designing interventions for causal discovery when the goal is to learn a portion of the edges in the causal graph instead of all of them.

In this work, we study the problem of designing the set of minimum cost interventions for identifying a specific causal effect, where intervening on each variable may have a different cost, and we are not necessarily allowed to intervene on every variable. Our contributions are as follows.

- We prove that finding a minimum cost intervention set for identifying a specific causal effect is NP-hard.
- We formulate the minimum cost intervention problem in terms of a minimum hitting set problem, and propose an algorithm based on this formulation that can find the optimal solution to the minimum cost intervention problem. This algorithm can also be used to approximate the solution up to a logarithmic-factor[1].
- We propose several heuristic algorithms to solve the minimum cost intervention problem in polynomial time, and through empirical evaluations show that they achieve low-regret solutions in randomly generated causal graphs.

[1]The implementations of all the algorithms proposed in this work can be found at `https://github.com/SinaAkbarii/min_cost_intervention/tree/main`.

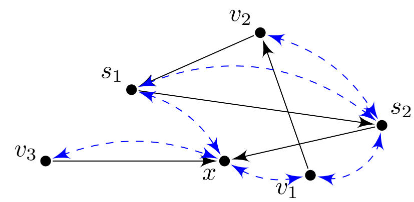


Figure 1. An example of a semi-Markovian graph. In this example, $\mathbf{pa}(x) = \{v_3, s_2\}$, $\mathbf{biD}(x) = \{v_3, s_1, v_1\}$, and $\mathbf{pa}^{\leftrightarrow}(x) = \{v_3\}$.

2. Terminology & Problem Description

We briefly introduce the notations used in this paper[2].

We use the *structural causal model* framework of (Pearl et al., 2000) in this work. We denote the corresponding semi-Markovian graph[3] over the observable variables V by $\mathcal{G}$ (Pearl et al., 2000; Tian & Pearl, 2002). A semi-Markovian graph has both directed and bidirected edges. A bidirected edge between two nodes implies that those nodes are affected by a hidden confounder. See Figure 1 for an example.

Since each vertex of $\mathcal{G}$ represents a random variable, we use the terms vertex and variable interchangeably. We use small letters for variables, capital letters for sets of variables, and bold letters for collections of subsets of variables, respectively. We utilize common graph-theoretic terms such as parents of a vertex x denoted by $\mathbf{pa}(x)$. The set of vertices that are connected to x via bidirected edges are denoted by $\mathbf{biD}(x)$, and $\mathbf{pa}^{\leftrightarrow}(x) = \mathbf{pa}(x) \cap \mathbf{biD}(x)$ denotes the intersection of parents of x and $\mathbf{biD}(x)$. For a set of variables X, $\mathbf{pa}(X)$ is defined as $\mathbf{pa}(X) = \cup_{x\in X}\mathbf{pa}(x) \setminus X$. $\mathbf{biD}(X)$ and $\mathbf{pa}^{\leftrightarrow}(X)$ are defined analogously. The induced subgraph of $\mathcal{G}$ over a subset $X \subseteq V$ is denoted by $\mathcal{G}_{[X]}$. The connected components of the edge induced subgraph of $\mathcal{G}$ over its bidirected edges are called *c-components* (aka *districts*) of $\mathcal{G}$, (Tian & Pearl, 2002). For example, the causal graph in Figure 1 consists of only one c-component. However, its induced subgraph over $\{s_1, x, v_2\}$ consists of two c-components $\{x, s_1\}$ and $\{v_2\}$.

We adopt the definition of interventional distributions using Pearl's $do()$ operator, i.e., $P(Y|do(X))$ denotes the causal effect of intervening on variables X on Y. We also denote by $Q[S]$ the causal effect of $do(V \setminus S)$ on S, that is, $Q[S] :=$

[2]We encourage the interested reader to see Appendix A for a comprehensive review of the terminology and more detailed definitions, as well as a few relevant known results in the literature.

[3]Also referred to as acyclic directed mixed graphs (ADMG)s in the literature (Evans & Richardson, 2014).

$P(S|do(V \setminus S))$.

Definition 1 (Identifiability). *We say a causal effect $P(Y|do(X))$ is identifiable in $\mathcal{G}$ given an observational distribution $P(V)$, if for any positive model M that is compatible with the causal graph $\mathcal{G}$ and $P_M(V) = P(V)$, $P_M(Y|do(X))$ is uniquely computable from $P_M(V)$.*

Analogously, for a given set of interventional distributions $\mathbf{P} = \{P(Y_1|do(X_1)), ..., P(Y_k|do(X_k))\}$, we say $P(S|do(T))$ is identifiable in the causal graph $\mathcal{G}$ from $\mathbf{P}$, if for any positive model M that is compatible with $\mathcal{G}$ and $P_M(Y_i|do(X_i)) = P(Y_i|do(X_i))$ for $1 \leq i \leq k$, $P_M(S|do(T))$ is uniquely computable from $\mathbf{P}$.

2.1. Problem Description

Let $\mathcal{G}$ be a semi-Markovian graph on the vertex set V along with a cost function $\mathbf{C} : V \to \mathbb{R}^{\geq 0}$, where $\mathbf{C}(x)$ for some $x \in V$ denotes the cost of intervening on variable x. With slight abuse of notation, we denote the cost of intervening on a set of variables $X \subseteq V$ by $\mathbf{C}(X)$. In this work, we assume that the intervention cost is additive, i.e., for a set $X \subseteq V$, the cost of intervening on X is $\mathbf{C}(X) := \sum_{x \in X} \mathbf{C}(x)$, and for a collection $\mathbf{X}$ of subsets of V, the cost of intervention on $\mathbf{X}$ is $\mathbf{C}(\mathbf{X}) := \sum_{X \in \mathbf{X}} \mathbf{C}(X)$. Moreover, we assume that there is no cost for observing a variable, i.e., $C(\emptyset) = 0$. Therefore, when intervening on set X, we have access to $Q[V \setminus X] = P(V \setminus X|do(X))$ at the cost of $C(X)$.

Remark 1. *In this setting, we can model a non-intervenable variable x by assigning the cost $\mathbf{C}(x) = \infty$.*

For a given causal graph $\mathcal{G}$ and disjoint subsets $S, T \subseteq V$, our goal is to find a collection $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ of subsets of V such that $P(S|do(T))$ is identifiable in $\mathcal{G}$ given $\{Q[V \setminus A_1], ..., Q[V \setminus A_m]\}$, and $C(\mathbf{A})$ is minimum. More precisely, let $\mathbf{ID}_\mathcal{G}(S, T)$ denote the set of all collections of subsets of V, e.g., $\mathbf{A} = \{A_1, A_2, ..., A_m\}$, where $A_i \subseteq V, 1 \leq i \leq m$, such that $P(S|do(T))$ is identifiable in $\mathcal{G}$ given $\{Q[V \setminus A_1], ..., Q[V \setminus A_m]\}$. Note that $|\mathbf{ID}_\mathcal{G}(S, T)| \leq 2^{2^{|V|}}$. Thus, the min-cost intervention design problem to identify $P(S|do(T))$ can be cast as the following optimization problem,

$$\mathbf{A}^*_{S,T} \in \arg \min_{\mathbf{A} \in \mathbf{ID}_\mathcal{G}(S,T)} \sum_{A \in \mathbf{A}} \mathbf{C}(A). \tag{1}$$

We say $\mathbf{A}^*_{S,T}$ is the min-cost intervention for identifying $P(S|do(T))$ in $\mathcal{G}$. Note that additional constraints or regularization terms can be added to target a specific min-cost intervention set within $\mathbf{ID}_\mathcal{G}(S, T)$.

It has been shown that $P(S|do(T))$ is identifiable in $\mathcal{G}$ if and only if $Q[\mathbf{Anc}_{\mathcal{G} \setminus T}(S)]$ is identifiable in $\mathcal{G}$, where $\mathbf{Anc}_{\mathcal{G} \setminus T}(S)$ are ancestors of S in $\mathcal{G}$ after deleting vertices T (Kivva et al., 2022; Lee et al., 2020; Jaber et al., 2019; Shpitser & Pearl, 2006). That is, $\mathbf{ID}_\mathcal{G}(S, T) = \mathbf{ID}_\mathcal{G}(\mathbf{Anc}_{\mathcal{G} \setminus T}(S), V \setminus \mathbf{Anc}_{\mathcal{G} \setminus T}(S))$. In other words, any causal query of the form $P(S|do(T))$ can be transformed into a causal query that is in the form of $Q[\cdot]$. Therefore, in what follows, we focus on the minimum-cost intervention problem for identifying causal queries of the form $Q[S] = P(S|do(V \setminus S))$. Throughout the rest of this work, we will assume $T = V \setminus S$ in Equation (1). In Section 3, we study the above problem when $\mathcal{G}_{[S]}$ is a single c-component. In Section 4, we generalize our results to an arbitrary subset S. We evaluate our proposed algorithms in terms of runtime and optimality in Section 5.

3. Single C-component Identification

The main challenge in solving the optimization problem in Equation (1) is that the number of elements in $\mathbf{ID}_\mathcal{G}(S, T)$ is possibly super-exponential. Throughout this section, we assume that S is a subset of variables in $\mathcal{G}$ such that $\mathcal{G}_{[S]}$ is a single c-component, unless stated otherwise. Under this assumption, we first show[4] in Theorem 1 that $\mathbf{ID}_\mathcal{G}(S, T)$ in Equation (1) can be replaced with a substantially smaller subset without changing the solution to the problem in (1). Next, we prove in Theorem 2 that even after this substitution, the min-cost intervention problem remains NP hard.

Lemma 1. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a single c-component. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets of V such that $A_\cup \cap S = \emptyset$, where $A_\cup := \cup_{i=1}^m A_i$. If $\mathbf{A} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S)$, then the singleton collection $\mathbf{A}_\cup = \{A_\cup\}$ also belongs to $\mathbf{ID}_\mathcal{G}(S, V \setminus S)$.*

Remark 2. *The cost of $\mathbf{A}_\cup$ in Lemma 1 is at most $C(\mathbf{A})$,*

$$\mathbf{C}(\mathbf{A}) = \sum_i \sum_{a \in A_i} \mathbf{C}(a) \geq \sum_{a \in A_\cup} \mathbf{C}(a) = \mathbf{C}(\mathbf{A}_\cup).$$

Next, we prove that for a given subset S where $\mathcal{G}_{[S]}$ is a c-component, the collection $\mathbf{A}^*_{S,V \setminus S}$ is singleton, that is, it contains exactly one intervention set.

Theorem 1. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets such that $\mathbf{A} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S)$ and $m > 1$. Then, there exists a subset $\tilde{A} \subseteq V$ such that $\tilde{\mathbf{A}} = \{\tilde{A}\} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S)$ and $\mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A})$.*

Theorem 1 indicates that when $\mathcal{G}_{[S]}$ is a c-component, the min-cost intervention problem in Equation 1 reduces to the problem of finding a single intervention set A^* such that $Q[S]$ is identifiable from $Q[V \setminus A^*]$. More formally, the optimization in (1) reduces to the following problem,

$$A^*_S \in \arg \min_{A \in \mathbf{ID}_1(S)} \sum_{a \in A} \mathbf{C}(a), \tag{2}$$

where $\mathbf{ID}_1(S)$ is the set of all subsets A of V such that $Q[S]$ is identifiable from $Q[V \setminus A]$. Note that $|\mathbf{ID}_1(S)| \leq 2^{|V|}$.

[4] All proofs are provided in Appendix B.

The following lemma states that intervening on the variables in S does not help in identifying $Q[S]$. As a result, we can further reduce the set $\mathbf{ID}_1(S)$ in (2). That is, we only consider all subsets A of V, such that $Q[S]$ is identifiable from $Q[V \setminus A]$ and $A \cap S = \emptyset$. This new set has at most $2^{|V|-|S|}$ elements. For the rest of this section, we discuss the solution to Equation (2).

Lemma 2. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. If $A \in \mathbf{ID}_1(S)$, then $A \cap S = \emptyset$.*

3.1. Hardness

In this section, we study the complexity of the min-cost intervention design problem in (2). We show that there exists a polynomial-time reduction from the Weighted Minimum Vertex Cover (WMVC) problem to the min-cost intervention problem. For the sake of completeness, we formally define the WMVC problem.

Definition 2 (WMVC)**.** *Given an undirected graph $\mathcal{H} = (V_{\mathcal{H}}, E_{\mathcal{H}})$ and a weight function $\omega : V_{\mathcal{H}} \to \mathbb{R}^{\geq 0}$, a vertex cover is a subset $A \subseteq V_{\mathcal{H}}$ such that A covers all the edges of $\mathcal{H}$, i.e., for any edge $\{x, y\} \in E_{\mathcal{H}}$, at least one of x or y is a member of A. The weighted minimum vertex cover problem's objective is to find a set A^* among all vertex covers that minimizes $\sum_{a\in A} \omega(a)$.*

WMVC is known to be NP-hard (Karp, 1972). Even finding an approximation within a factor of 1.36 to this problem is NP-hard (Dinur & Safra, 2005). In fact, there is no known polynomial-time algorithm to approximate WMVC problem within a constant factor less than two[5]. Indeed, WMVC remains NP-hard even for bounded-degree graphs (Garey et al., 1974). The following theorem shows that all these statements also hold for the min-cost intervention problem.

Theorem 2. *WMVC problem is reducible to a min-cost intervention problem in polynomial time.*

Remark 3. *The unweighted version of WMVC problem (i.e., when the weight function is given by $\omega(\cdot) = 1$) can be reduced to a minimum-cost intervention problem with the constant cost function $\mathbf{C}(\cdot) = 1$ in polynomial time.*

Theorem 2 states that any algorithm that solves the min-cost intervention problem, can also solve WMVC with a polynomial overhead. It immediately follows that min-cost intervention is NP-hard, and it is hard to approximate within a constant factor less than 1.36. On the other hand, as with any other NP-hard problem, certain instances of the min-cost intervention problem can be solved in polynomial-time. An interesting group of such instances are discussed in Appendix C. Despite being restrictive, these special cases might provide useful insights for finding efficient algorithms in more general settings. Naturally, Theorem 2 implies that the algorithms proposed in this paper can aid to solve some other problems in the NP class.

[5]Factor 2 approximation algorithms appear in (Garey & Johnson, 1979; Papadimitriou & Steiglitz, 1998).

3.2. Minimum Hitting Set Formulation

In this section, we propose a formulation of the min-cost intervention problem in terms of yet another NP-hard problem known as the minimum-weight hitting set (MWHS) problem. This formulation will allow us to find algorithms to solve or approximate our problem in the later sections.

Definition 3 (MWHS)**.** *Let $V = \{v_1, ..., v_n\}$ be a set of objects along with a weight function $\omega : V \to \mathbb{R}^{\geq 0}$. Given a collection of subsets of V such as $\mathbf{F} = \{F_1, ..., F_k\}$, $F_i \subseteq V$, $1 \leq i \leq k$, a hitting set for $\mathbf{F}$ is a subset $A \subseteq V$ such A hits all the sets in $\mathbf{F}$, i.e., for any $1 \leq i \leq k$, $A \cap F_i \neq \emptyset$. The weighted minimum hitting set problem's objective is to find a set A^* among all hitting sets that minimizes $\sum_{a\in A} \omega(a)$.*

It is known that special structures, called *hedges* that are formed for $Q[S]$ in $\mathcal{G}$ prevent the identifiability of the causal effect $Q[S]$ (Shpitser & Pearl, 2006). On the other hand, intervening on a vertex of a hedge allows us to eliminate it from the graph. Hence, the problem of identifying $Q[S]$ is equivalent to finding a subset of vertices that hits all the hedges formed for $Q[S]$. In other words, the min-cost intervention problem can be reformulated as a MWHS problem. For simplicity, here, we use a slightly modified definition of a hedge. In Appendix A, we show that it is equivalent to the original definition in (Shpitser & Pearl, 2006).

Definition 4. *(Hedge) Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices such that $\mathcal{G}_{[S]}$ is a c-component. A subset F is a hedge formed for $Q[S]$ in $\mathcal{G}$ if $S \subsetneq F$, F is the set of ancestors of S in $\mathcal{G}_{[F]}$, and $\mathcal{G}_{[F]}$ is a c-component.*

As an example, let $S = \{s_1, s_2\}$ in the causal graph of Figure 1. In this case, $\mathcal{G}_{[S]}$ is a c-component and $\{s_1, s_2, v_1, v_2\}$ and $\{s_1, s_2, v_2\}$ are two hedges formed for $Q[S]$.

Using the result of (Shpitser & Pearl, 2006), the following Lemma connects the minimum-cost intervention problem to the minimum-weight hitting set problem.

Lemma 3. *Let $\mathcal{G}$ be a semi-Markovian graph with vertex set V, along with a cost function $\mathbf{C} : V \to \mathbb{R}^{\geq 0}$. Let S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Suppose the set of all hedges formed for $Q[S]$ in $\mathcal{G}$ is $\{F_1, ..., F_m\}$. Then A^*_S is a solution to Equation (2) if and only if it is a solution to the MWHS problem for the sets $\{F_1 \setminus S, ..., F_m \setminus S\}$, with the weight function $\omega(\cdot) := \mathbf{C}(\cdot)$.*

Lemma 3 suggests that designing an intervention to identify $Q[S]$ can be cast as finding a set that intersects (hits) with all the hedges formed for $Q[S]$. A brute-force algorithm to find the minimum-cost intervention (Equation (2)) is then to first enumerate all hedges formed for $Q[S]$ in $\mathcal{G}$ and solve the corresponding hitting set problem.

Solving MWHS, which itself is equivalent to the set cover problem, is known to be NP-hard (Karp, 1972; Bernhard & Vygen, 2008). However, there exist greedy algorithms that can approximate the optimal solution up to a logarithmic factor (Johnson, 1974; Chvatal, 1979), which has been shown to be optimum in the sense that they achieve the best approximation ratio (Feige, 1998). Another approach for tackling MWHS is via linear programming relaxation which achieves the similar approximation ratio as the greedy ones (Lovász, 1975). But even in the case that we use an approximation algorithm for the hitting set formulation of the min-cost intervention problem, the task of enumerating all hedges formed for $Q[S]$ in $\mathcal{G}$ requires exponential number of computations in terms of number of the variables.

3.3. Properties of A_S^*

In Section 3.1, we proved that the min-cost intervention design problem in (2) is NP-hard. Herein, we shall study certain properties of the solution A_S^* that allow us to reduce the complexity of solving (2). We begin with characterizing a set of variables that we *must* intervene upon to identify $Q[S]$.

Recall that $\textbf{pa}^{\leftrightarrow}(S)$ is the set of parents of S that have a bidirected edge to a variable in S. Note that for a given set S, we can construct $\textbf{pa}^{\leftrightarrow}(S)$ in linear time. The following Lemma indicates that $Q[S]$ is not identifiable unless all of the variables in $\textbf{pa}^{\leftrightarrow}(S)$ are intervened upon.

Lemma 4. *Let $\mathcal{G}$ be a semi-Markovian graph with the vertex set V, and for $S \subseteq V$, let $\mathcal{G}_{[S]}$ be a c-component. For any subset $A \subseteq V$, if $A \in \mathbf{ID}_1(S)$, then $\textbf{pa}^{\leftrightarrow}(S) \subseteq A$.*

As a counterpart to Lemma 4, below, we characterize a subset of vertices that do not belong to A_S^*.

Definition 5 (Hedge hull)**.** *Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices such that $\mathcal{G}_{[S]}$ is a c-component. The union of all hedges formed for $Q[S]$ is called hedge hull of S and denoted by $Hhull(S, \mathcal{G})$.*

If $\mathcal{G}_{[S]}$ is not a c-component, it can be uniquely partitioned into maximal c-components (Tian & Pearl, 2002). Let $S_1, ..., S_k$ be the partition of S such that $\mathcal{G}_{[S_1]}, ..., \mathcal{G}_{[S_k]}$ are the maximal c-components of $\mathcal{G}_{[S]}$. We define $Hhull(S, \mathcal{G})$ as $Hhull(S, \mathcal{G}) = \bigcup_{i=1}^{k} Hhull(S_i, \mathcal{G})$.

Lemma 5. *Consider A_S^* in Equation* (2)*, then $A_S^* \subseteq Hhull(S, \mathcal{G}) \setminus S$.*

For a given subset S and a semi-Markovian graph $\mathcal{G}$, Lemmas 4 and 5 bound the solution to the min-cost intervention problem as $\textbf{pa}^{\leftrightarrow}(S) \subseteq A_S^* \subseteq Hhull(S, \mathcal{G})$.

In Algorithm 1, we propose a method to construct the hedge hull of a given subset S. Lines 3 and 4 of this algorithm can be performed via *depth first search* (DFS) algorithm, which is quadratic in the number of vertices in the worst-case scenario[6]. On the other hand, the while loop of line 2 can run at most $|V|$ times in the worst case (as long as $F_2 \neq F$, at least one vertex will be eliminated from F.) Hence, the complexity of this algorithm is[7] $\mathcal{O}(|V|^3)$.

Algorithm 1 Find $Hhull(S, \mathcal{G})$, $\mathcal{G}_{[S]}$ is a c-component.

```
1: Initialize F ← V
2: while True do
3:    F_1 ← connected component of S via bidirected
      edges in G_[F]
4:    F_2 ← ancestors of S in G_[F_1]
5:    if F_2 ≠ F then
6:       F ← F_2
7:    else
8:       break
9: return F
```

Lemma 6. *Given a semi-Markovian graph $\mathcal{G}$ over V and a subset $S \subseteq V$ such that $\mathcal{G}_{[S]}$ is a c-component, Algorithm 1 returns $Hhull(S, \mathcal{G})$ in $\mathcal{O}(|V|^3)$.*

Next theorem summarizes the results of this Section.

Theorem 3. *Let S be a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. Then, A_S^* is a solution to* (2) *if and only if both $\textbf{pa}^{\leftrightarrow}(S) \subseteq A_S^*$ and $A_S^* \setminus \textbf{pa}^{\leftrightarrow}(S)$ is a min-cost intervention to identify $Q[S]$ in $\mathcal{G}_{[H]}$, where*

$$H := Hhull(S, \mathcal{G}_{[V \setminus \textbf{pa}^{\leftrightarrow}(S)]}). \tag{3}$$

This result suggests that solving (2) can be done by first identifying $\textbf{pa}^{\leftrightarrow}(S)$, and then solving a reduced size min-cost intervention problem to identify $Q[S]$ in $\mathcal{G}_{[H]}$, where H is given in Equation (3). Note that all the minimal hedges of S in $\mathcal{G}_{[H]}$ can be enumerated in $\mathcal{O}(2^{(|H|-|S|)})$. Therefore, if $|H|$ is small, the hedge enumeration task of the brute-force approach in Section 3.2 can be done efficiently. However, the performance of this method deteriorates as the size of H increases. Next, we propose an algorithm that circumvents the hedge enumeration task to solve the min-cost intervention problem more efficiently.

[6]Lines 3 and 4 can also be swapped, as the order in which we execute them does not affect the output.

[7]To be more precise, DFS takes time $\mathcal{O}(|V| + |E|)$, where $|E|$ is the number of edges. Therefore, Alg. 1 runs in time $\mathcal{O}(|V|^2 + |V| \cdot |E|)$.

3.4. Exact Algorithmic Solution to Min-cost Intervention Problem

In this section, we propose an algorithm that can be used both to exactly solve the min-cost intervention problem and to approximate it within a logarithmic factor.

As we discussed earlier, the min-cost intervention problem can be formulated as a combination of two tasks: enumerating the hedge structures and solving a minimum hitting set for the hedges. Although minimum hitting set problem can be solved with polynomial-time approximation algorithms, enumerating all hedges requires exponential computational complexity. To reduce this complexity, we propose Algorithm 2, that avoids enumerating all hedges formed for $Q[S]$ by utilizing the notion of minimality defined below, and Theorem 3. We will next explain this.

Definition 6 (Minimal hedge)**.** *A hedge F formed for $Q[S]$ in $\mathcal{G}$ is said to be minimal if no subset of F (clearly excluding F) is a hedge formed for $Q[S]$ in $\mathcal{G}$.*

Algorithm 2 Min-cost intervention$(S, \mathcal{G})$.

1: $\mathbf{F} \leftarrow \emptyset, \quad H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]})$
2: **if** $H = S$ **then**
3:    **return** $\mathbf{pa}^{\leftrightarrow}(S)$
4: **while** True **do**
5:    **while** True **do**
6:       $a \leftarrow \arg\min_{a \in H \setminus S} \mathbf{C}(a)$
7:       **if** $Hhull(S, \mathcal{G}_{[H \setminus \{a\}]}) = S$ **then**
8:          $\mathbf{F} \leftarrow \mathbf{F} \cup \{H\}$
9:          **break**
10:       **else**
11:          $H \leftarrow Hhull(S, \mathcal{G}_{[H \setminus \{a\}]})$
12:    $A \leftarrow$ solve min hitting set for $\{F \setminus S | F \in \mathbf{F}\}$
13:    **if** $A \cup \mathbf{pa}^{\leftrightarrow}(S) \in \mathbf{ID_1}(S)$ **then**
14:       **return** $(A \cup \mathbf{pa}^{\leftrightarrow}(S))$
15:    $H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus (A \cup \mathbf{pa}^{\leftrightarrow}(S))]})$

As an example, in Figure 1, $Q[\{s_1, s_2\}]$ has two hedges: $\{s_1, s_2, v_1, v_2\}$ and $\{s_1, s_2, v_2\}$. In this case, $\{s_1, s_2, v_2\}$ is a minimal hedge. Clearly, every non-minimal hedge formed for $Q[S]$ has a subset which is a minimal hedge. Therefore, hitting all the *minimal* hedges would suffice to identify $Q[S]$. As a result, for hedge enumeration, whenever we find a subset F that is a hedge formed for $Q[S]$, it is not necessary to consider any super-set of F.

Algorithm 2 begins with identifying $\mathbf{pa}^{\leftrightarrow}(S)$ and the subset H given in (3). The main idea of this algorithm is to discover a subset of the hedges formed for $Q[S]$ in $\mathcal{G}$ denoted by $\mathbf{F}$, such that the minimum hitting set solution for $\mathbf{F}$ is exactly the solution to the original min-cost intervention problem. It constructs $\mathbf{F}$ iteratively. To this end, within the inner loop (lines 6-12), it selects a vertex a in $H \setminus S$ with the minimum cost, and removes a from H (resolves the hedge H). If this hedge elimination makes $Q[S]$ identifiable (i.e., $Hhull(S, \mathcal{G}_{[H \setminus \{a\}]}) = S$), it updates $\mathbf{F}$ in line 9. Otherwise, it updates H by $Hhull(S, \mathcal{G}_{H \setminus \{a\}})$ in line 12 using Algorithm 1. The reason for updating $\mathbf{F}$ only when $Q[S]$ becomes identifiable is that the hedge discovered in the last step of the inner loop H is a subset of all the hedges discovered earlier. Therefore, hitting (eliminating) H, also hits all its super-sets.

At the end of the inner loop, it solves a minimum hitting set problem for the constructed $\mathbf{F}$ to find A in line 13. If $A \cup \mathbf{pa}^{\leftrightarrow}(S) \in \mathbf{ID_1}(S)$, the algorithm terminates and outputs $A \cup \mathbf{pa}^{\leftrightarrow}(S)$ as the optimal intervention set. Otherwise, it updates H using Algorithm 1 in line 16 and repeats the outer loop by going back to line 5 to discover new hedges formed for $Q[S]$.

In the worst-case scenario, Algorithm 2 requires exponential number of iterations to form $\mathbf{F}$. However, as illustrated in our empirical evaluations in Appendix F, the algorithm often finds the solution to the min-cost intervention after only a few number of iterations. This is to say, in practice, discovering only a few hedges and solving the hitting set problem for them suffices to solve the original min-cost intervention problem.

Lemma 7. *Let $\mathcal{G}$ be a semi-Markovian graph and $S \subseteq V$. Algorithm 2 returns an optimal solution to* (2).

It is noteworthy that this result holds even if S is not a c-component. In other words, Algorithm 2 always returns an optimal solution in $\mathbf{ID}_1(S)$. We will use this result in Section 4 to introduce an algorithm for the general setting in which S is an arbitrary subset of variables.

Approximation. Note that the minimum hitting set problem in line 13 can be solved approximately using a greedy algorithm (Johnson, 1974; Chvatal, 1979), which guarantees a logarithmic-factor approximation[8]. In this case, if polynomially many hedges are discovered before the algorithm stops[9], Algorithm 2 returns a logarithmic-factor approximation of the solution in polynomial time.

3.5. Heuristic Algorithms

The algorithm discussed in the previous Section provides an exact solution for finding the minimum cost intervention. However, it has an exponential runtime in the worst case. Herein, we develop and present two heuristic algorithms to approximate the solution to the min-cost intervention

[8]See Appendix E for further details.

[9]We propose a slightly modified version of Algorithm 2 in Appendix E with lower number of calls to the hitting set solver.

problem in polynomial time. In Section 5, we evaluate the performance of these algorithms in terms of their runtimes and the optimality of their solutions. The detailed analysis of these heuristic algorithms are provided in Appendix D. It is noteworthy that these two algorithms utilize the result of Theorem 3, i.e., they initiate with identifying $\mathbf{pa}^{\leftrightarrow}(S)$, H in (3), and then find a min-cost intervention set that identifies $Q[S]$ in $\mathcal{G}_{[H]}$. Theses two algorithms approximate the min-cost intervention problem via a minimum weight vertex cut (a.k.a. vertex separator) problem.

Definition 7. *(Minimum weight vertex cut) Let $\mathcal{H}$ be a (un)directed graph over the vertices V, with a weight function $\omega : V \to \mathbb{R}^{\geq 0}$. For two non-adjacent vertices $x, y \in V$, a subset $A \subset V \setminus \{x, y\}$ is said to be a vertex cut for $x - y$, if there is no (un)directed path that connects x to y in $\mathcal{H}_{V \setminus A}$. The objective of minimum weight vertex cut problem is to identify a vertex cut for $x - y$ that minimizes $\sum_{a \in A} \omega(a)$.*

Min-weight vertex cut problem can be solved in polynomial time by, for instance, casting it as a max-flow problem[10] and then using algorithms such as Ford-Fulkerson, Edmonds-Karp, or push-relabel algorithm (Ford & Fulkerson, 1956; Edmonds & Karp, 1972; Goldberg & Tarjan, 1988).

Heuristic Algorithm 1: For a given graph $\mathcal{G}$ and a subset $S \subseteq V$, this algorithm builds an undirected graph $\mathcal{H}$ with the vertex set $H \cup \{x, y\}$, where H is given in (3), and x and y are two auxiliary vertices. For any pair of vertices $\{v_1, v_2\} \in H$, if v_1 and v_2 are connected with a bidirected edge in $\mathcal{G}$, they will be connected in $\mathcal{H}$. Vertex x is connected to all the vertices in $\mathbf{pa}(S) \cap H$, and y is connected to all vertices in S. The output of the algorithm is the minimum weight vertex cut for $x - y$, with the weight function $\omega(\cdot) := \mathbf{C}(\cdot)$. Algorithm 7 in Appendix D presents the pseudo code for this procedure. Next result shows that intervening on the output set of this algorithm will identify $Q[S]$, although this set is not necessarily minimum-cost.

Lemma 8. *Let $\mathcal{G}$ be a semi-Markovian graph on V and S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Heuristic Algorithm 1 returns an intervention set A in $\mathcal{O}(|V|^3)$ such that $A \in \mathbf{ID}_1(S)$.*

Heuristic Algorithm 2: Given a graph $\mathcal{G}$ and a subset S, this algorithm builds a directed graph $\mathcal{J}$ as follows: the vertex set is $H \cup \{x, y\}$, where H is given in (3), and x and y are two auxiliary vertices. For any pair of vertices $\{v_1, v_2\} \in H$, if v_1 is a parent of v_2 in $\mathcal{G}$, then v_1 will be a parent of v_2 in $\mathcal{J}$. Vertex x is added to the parent set of all vertices in $\mathbf{biD}(S) \cap H$, and all vertices of S are added to the parent set of y. The output of this algorithm is the minimum weight vertex cut for $x - y$, with the weight function $\omega(\cdot) := \mathbf{C}(\cdot)$. Algorithm 8 in Appendix D summarizes this procedure. The following result indicates that intervening on the output set of this algorithm identifies $Q[S]$.

Lemma 9. *Let $\mathcal{G}$ be a semi-Markovian graph on V and S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Heuristic Algorithm 2 returns an intervention set A in $\mathcal{O}(|V|^3)$ such that $A \in \mathbf{ID}_1(S)$.*

A major difference between the two heuristic algorithms is that Algorithm 2 solves a minimum vertex cut on a directed graph, whereas Algorithm 1 solves the same problem on an undirected graph. Since the equivalent max-flow problem is easier to solve on directed graphs, Algorithm 2 is preferred, unless the directed edges of $\mathcal{G}$ are considerably denser than its bidirected edges. We propose another algorithm which uses a greedy approach to solve the min-cost intervention problem and discuss its complexity in Appendix D. This greedy algorithm is preferable to the two aforementioned algorithms in certain special settings. Additionally, we propose a polynomial-time post-process in Appendix D to improve the solution returned by our three heuristic algorithms.

4. General Subset Identification

So far we have discussed the min-cost intervention design problem for subset S, where the induced subgrah $\mathcal{G}_{[S]}$ is a c-component. In this section, we study the general case in which S is an arbitrary subset of variables and show that the min-cost intervention design problem for S requires solving a set of instances of the problem for single c-component.

The main challenge in the general case is that Theorem 1 is no longer valid. Thus, the min-cost intervention design problem in (1) cannot be reduced to (2). As an example, consider Figure 2. In this causal graph, the minimum-cost intervention to identify $Q[S]$ for $S := \{s_1, s_2, s_3\}$, is $\mathbf{A}^*_{S,V\setminus S} = \{\{s_1\}, \{s_2\}\}$ with the cost $\mathbf{C}(s_1) + \mathbf{C}(s_2) = 2$. However, any singleton intervention that can identify $Q[S]$, i.e., $A \in \mathbf{ID}_1(S)$ has a cost of at least 10. More importantly, the union of the sets in $\mathbf{A}^*_{S,V\setminus S}$, i.e., $\{s_1, s_2\}$ does not belong to $\mathbf{ID}_1(S)$. In other words, intervening on $\{s_1, s_2\}$ does not identify $Q[S]$ (Lemma 2).

In many applications, it is reasonable to assume that in order to identify $Q[S] = P(S|do(V \setminus S))$, intervening on elements of S is not desirable. In other words, $\mathbf{C}(s) = \infty$, for all $s \in S$. Under this assumption, we show that instances similar to Figure 2 cannot occur and results analogous to Theorem 1 can be established.

Theorem 4. *Suppose S is a subset of variables such that $\mathbf{C}(s) = \infty$ for any $s \in S$. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets such that $\mathbf{A} \in \boldsymbol{ID}_{\mathcal{G}}(S, V \setminus S)$ and $m > 1$. Then there exists a singleton intervention $\tilde{A}$ such that $\tilde{\mathbf{A}} = \{\tilde{A}\} \in \boldsymbol{ID}_{\mathcal{G}}(S, V \setminus S)$ and $\mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A})$.*

Theorem 4 implies that the general problem can be solved

[10] See Appendix D for details.

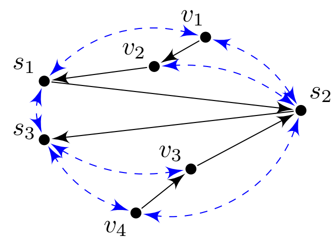


Figure 2. An example where the optimal interventions collection is not a singleton, i.e., a solution to (2) is not a solution to (1). In this example, the cost of intervening on each of $\{s_1, s_2, s_3\}$ is 1, whereas the cost of intervening on each of $\{v_1, v_2, v_3, v_4\}$ is 5.

Algorithm 3 General algorithm$(S, \mathcal{G})$.

1: $\mathbf{A}^* \leftarrow null$, minCost $\leftarrow \infty$
2: $\{S_1, ..., S_k\} \leftarrow$ maximal c-components of $\mathcal{G}_{[S]}$
3: **for** any partition of $\{S_1, ..., S_k\}$ as $S^{(1)}, ..., S^{(t)}$ **do**
4: $\quad \mathbf{A} \leftarrow \{\}$, cost $\leftarrow 0$
5: $\quad$ **for** i from 1 to t **do**
6: $\quad\quad A_i \leftarrow$ min-cost intervention set in $\mathbf{ID_1}(\underline{S}^{(i)})$
7: $\quad\quad \mathbf{A} \leftarrow \mathbf{A} \cup \{A_i\}$, cost $\leftarrow$ cost $+\mathbf{C}(A_i)$
8: $\quad$ **if** cost $<$ minCost **then**
9: $\quad\quad \mathbf{A}^* \leftarrow \mathbf{A}$, minCost $\leftarrow$ cost
10: **return** $\mathbf{A}^*$

exactly analogous to the case where $\mathcal{G}_{[S]}$ is a c-component.

We now turn to proposing an exact solution to the min-cost intervention problem in (1) for general case. Let $S_1, ..., S_k$ be subsets of S such that $\bigcup_i S_i = S$ and $\mathcal{G}_{[S_1]}, ..., \mathcal{G}_{[S_k]}$ are the maximal[11] c-components of $\mathcal{G}_{[S]}$. It is known that $Q[S]$ is identifiable in $\mathcal{G}$ if and only if $Q[S_i]$s are identifiable in $\mathcal{G}$ for all $1 \leq i \leq k$ (Tian & Pearl, 2002). Based on this observation, we show the following result.

Lemma 10. *Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices. Suppose $\mathbf{A}^*_{S,V\setminus S}$ is a min-cost interventions collection to identify $Q[S]$ in $\mathcal{G}$. If $\mathcal{G}_{[S_j]}$ is a maximal c-component of $\mathcal{G}_{[S]}$, then there exists $A \in \mathbf{A}^*_{S,V\setminus S}$ such that $A \in \mathbf{ID_1}(S_j)$.*

According to Lemma 10, for each maximal c-component of $\mathcal{G}_{[S]}$ such as $\mathcal{G}_{[S_j]}$, one intervention set suffices to identify $Q[S_j]$. As a result, we can partition the maximal c-components of $\mathcal{G}_{[S]}$ such that each partition is identified using a singleton intervention set. Therefore, to solve the min-cost intervention problem for $Q[S]$, for every possible partitioning of the maximal c-components of $\mathcal{G}_{[S]}$, we find a collection of singletons that can identify the elements in that partition. The min-cost intervention collection to identify $Q[S]$ in $\mathcal{G}$ is then the collection with the lowest cost.

More precisely, let $S^{(1)}, ..., S^{(t)}$ denote a partitioning of $\{S_1, ..., S_k\}$. That is, for each j, $S^{(j)}$ is a subset of set $\{S_1, ..., S_k\}$, $S^{(j)} \cap S^{(i)} = \emptyset$ for $i \neq j$, and $\bigcup_{j=1}^{t} S^{(j)} = \{S_1, ..., S_k\}$. Furthermore, we denote the set of all vertices in partition $S^{(j)}$ by $\underline{S}^{(j)}$. As an example, in Figure 2, set $S = \{s_1, s_2, s_3\}$ consists of two maximal c-components $\mathcal{G}_{[S_1]}$ and $\mathcal{G}_{[S_2]}$, where $S_1 = \{s_1, s_3\}$ and $S_2 = \{s_2\}$. There are two different ways to partition $\{S_1, S_2\}$. One is $S^{(1)} = \{S_1, S_2\}$. The other is $S^{(1)} = \{S_1\}$ and $S^{(2)} = \{S_2\}$. For the first partition, we have $\underline{S}^{(1)} = \{s_1, s_2, s_3\}$. Similarly, the second partition will result in $\underline{S}^{(1)} = \{s_1, s_3\}$ and $\underline{S}^{(2)} = \{s_2\}$.

In order to solve the min-cost intervention problem for $Q[S]$, for every possible partitioning of the maximal c-components of $\mathcal{G}_{[S]}$ such as $S^{(1)}, ..., S^{(t)}$, we solve for the min-cost intervention set A_j to identify $Q[\underline{S}^{(j)}]$ for every $1 \leq j \leq t$, and form the intervention collection $\mathbf{A} = \{A_1, ..., A_t\}$. The aggregate cost for this collection of interventions is given by $\mathbf{C}(\mathbf{A})$. The min-cost intervention collection to identify $Q[S]$ in $\mathcal{G}$ is the one with minimum aggregate cost. Algorithm 3 summarizes this procedure. Although the number of partitions of a set, known as the Bell number can grow super-exponentially, since the number of maximal c-components of the set S considered in practical problems is small, the runtime of Algorithm 3 is expected to be manageable.

Having partitioned the maximal c-components of $\mathcal{G}_{[S]}$, the only remaining challenge is to perform line (6) of Algorithm 3, i.e., to find the min-cost singleton intervention set to identify the maximal c-components within a partition. Note that the hitting set formulation introduced in Section 3.2, i.e., Algorithm 2 is a valid approach to find such a singleton - the soundness of this algorithm was shown in Lemma 7.

Proposition 1. *Given a semi-Markovian graph $\mathcal{G}$ and a subset S of its vertices, Algorithm 3 with Algorithm 2 used as a subroutine in line (6) returns an optimal solution to the min-cost interventions collection to identify $Q[S]$ in $\mathcal{G}$.*

5. Evaluations

For evaluation, we generated the causal graphs using the Erdos-Renyi generative model (Erdős & Rényi, 1960) as follows. For a given number of vertices n, we fixed a causal order over the vertices. Then, directed edges were sampled with probability $p = 0.35$ and bidirected edges were sampled with probability $q = 0.25$ between the vertices, mutually independently. The set S was selected randomly among the last 5% of the vertices in the causal order such that $\mathcal{G}_{[S]}$ is a c-component. Intervention costs of vertices were chosen independently at random from $\{1, 2, 3, 4\}$. See Appendix F for further details of the evaluation setup.

[11] See Appendix A for details.

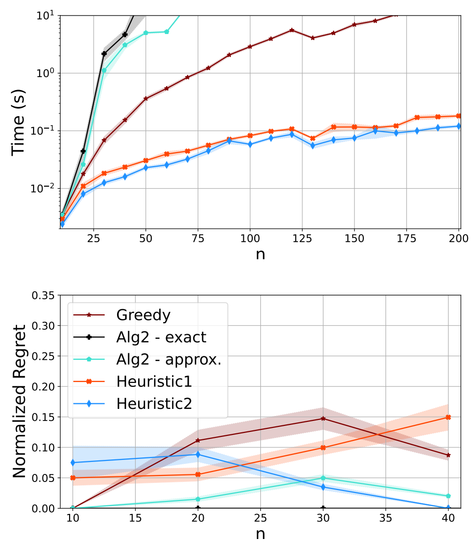


Figure 3. Evaluation of our algorithms in terms of runtime (Top) and normalized regret (Bottom). n represents the number of vertices in the causal graph, Alg2-exact and Alg2-approx. solve the minimum hitting set exactly and approximately, respectively, and the greedy algorithm is presented in Appendix D. Alg2-exact obtains zero regret.

For various n, we sampled different causal graphs and different subsets S using the above procedure and ran our algorithms[12] on them to find the min-cost intervention set for identifying $Q[S]$. Our performance measures are runtime and normalized regret. Normalized regret of a given subset A is defined by $(\mathbf{C}(A) - \mathbf{C}^*)/\mathbf{C}^*$, where $\mathbf{C}^*$ denotes the optimal min-cost solution. The results are depicted in Figure 3. Each curve and its confidence interval is obtained by averaging over 40 trials. As illustrated in Figure 3, our proposed heuristic algorithms achieve negligible regret in most of the cases while their runtime are considerably faster than the exact algorithm, i.e., Algorithm 2. It is noteworthy that the regret is not necessarily a monotone function of n, and it depends on the structure of the causal graph and intervention costs. For further evaluations of our algorithms (e.g., their sensitivities with respect to p and q), see Appendix F.

6. Concluding Remarks

We discussed the problem of designing the minimum cost intervention for causal effect identification in this paper. We established the NP-hardness of this problem by relating it to two well-known NP-hard problems: minimum vertex cover and hitting set. We proposed an algorithm based on the hitting set problem to solve the min-cost intervention problem exactly and proposed several heuristic algorithms to design an intervention in polynomial time.

[12] `https://github.com/SinaAkbarii/min_cost_intervention/tree/main`

Acknowledgements

This research was in part supported by the Swiss National Science Foundation under NCCR Automation, grant agreement 51NF40_180545 and Swiss SNF project 200021_204355 /1.

References

Addanki, R., Kasiviswanathan, S., McGregor, A., and Musco, C. Efficient intervention design for causal discovery with latents. In *International Conference on Machine Learning*, pp. 63–73. PMLR, 2020.

Addanki, R., McGregor, A., and Musco, C. Intervention efficient algorithms for approximate learning of causal graphs. In *Algorithmic Learning Theory*, pp. 151–184. PMLR, 2021.

Agrawal, R., Squires, C., Yang, K., Shanmugam, K., and Uhler, C. Abcd-strategy: Budgeted experimental design for targeted causal structure discovery. In *the 22nd International Conference on Artificial Intelligence and Statistics*, pp. 3400–3409. PMLR, 2019.

Bareinboim, E. and Pearl, J. Causal inference by surrogate experiments: z-identifiability. *arXiv preprint arXiv:1210.4842*, 2012.

Bernhard, K. and Vygen, J. Combinatorial optimization: Theory and algorithms. *Springer, Third Edition, 2005.*, 2008.

Chvatal, V. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4(3):233–235, 1979.

Dinur, I. and Safra, S. On the hardness of approximating minimum vertex cover. *Annals of mathematics*, pp. 439–485, 2005.

Edmonds, J. and Karp, R. M. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972.

Erdős, P. and Rényi, A. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:17–61, 1960.

Evans, R. J. and Richardson, T. S. Markovian acyclic directed mixed graphs for discrete data. *The Annals of Statistics*, 42(4):1452–1482, 2014.

Feige, U. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4):634–652, 1998.

Ford, L. R. and Fulkerson, D. R. Maximal flow through a network. *Canadian journal of Mathematics*, 8:399–404, 1956.

Foster, E. M. Causal inference and developmental psychology. *Developmental psychology*, 46(6):1454, 2010.

Gangl, M. Causal inference in sociological research. *Annual review of sociology*, 36:21–47, 2010.

Garey, M. R. and Johnson, D. S. *Computers and intractability: a guide to the theory of NP-completeness*. W. H. Freeman, 1979.

Garey, M. R., Johnson, D. S., and Stockmeyer, L. Some simplified np-complete problems. In *Proceedings of the sixth annual ACM symposium on Theory of computing*, pp. 47–63, 1974.

Ghassami, A., Salehkaleybar, S., Kiyavash, N., and Bareinboim, E. Budgeted experiment design for causal structure learning. In *International Conference on Machine Learning*, pp. 1724–1733. PMLR, 2018.

Goldberg, A. V. and Tarjan, R. E. A new approach to the maximum-flow problem. *Journal of the ACM (JACM)*, 35(4):921–940, 1988.

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

Hoover, K. D. The logic of causal inference: Econometrics and the conditional analysis of causation. *Economics & Philosophy*, 6(2):207–234, 1990.

Huang, Y. and Valtorta, M. Pearl's calculus of intervention is complete. In *Uncertainty in Artificial Intelligence*, volume 22, pp. 437–444, 2006.

Jaber, A., Zhang, J., and Bareinboim, E. Causal identification under markov equivalence: Completeness results. In *International Conference on Machine Learning*, pp. 2981–2989. PMLR, 2019.

Jensen, F., Kjærulff, U., Olesen, K., and Pedersen, J. An expert system for control of waste water treatment—a pilot project. Technical report, Technical report, Judex Datasystemer A/S, Aalborg, 1989. In Danish, 1989.

Johnson, D. S. Approximation algorithms for combinatorial problems. *Journal of computer and system sciences*, 9(3): 256–278, 1974.

Kandasamy, S., Bhattacharyya, A., and Honavar, V. G. Minimum intervention cover of a causal graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 2876–2885, 2019.

Karp, R. M. Reducibility among combinatorial problems. In *Complexity of computer computations*, pp. 85–103. Springer, 1972.

Kivva, Y., Mokhtarian, E., Etesami, J., and Kiyavash, N. Revisiting the general identifiability problem. In *The 38th Conference on Uncertainty in Artificial Intelligence*, 2022.

Konig, D. Graphok es matrixok (hungarian)[graphs and matrices]. *Matematikai és Fizikai Lapok*, 38:116–119, 1931.

Kristensen, K. and Rasmussen, I. A decision support system for mechanical weed control in malting barley. In *Proceedings of the First European Conference on Information Technology in Agriculture*, pp. 447–452, 1997.

Lee, S., Correa, J. D., and Bareinboim, E. General identifiability with arbitrary surrogate experiments. In *Uncertainty in Artificial Intelligence*, pp. 389–398. PMLR, 2020.

Lindgren, E. M., Kocaoglu, M., Dimakis, A. G., and Vishwanath, S. Experimental design for cost-aware learning of causal graphs. *arXiv preprint arXiv:1810.11867*, 2018.

Lovász, L. On the ratio of optimal integral and fractional covers. *Discrete mathematics*, 13(4):383–390, 1975.

Murnane, R. J. and Willett, J. B. *Methods matter: Improving causal inference in educational and social science research*. Oxford University Press, 2010.

Papadimitriou, C. H. and Steiglitz, K. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

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

Pearl, J. The do-calculus revisited. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pp. 3–11, 2012.

Pearl, J. et al. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19, 2000.

Robins, J. A graphical approach to the identification and estimation of causal parameters in mortality studies with sustained exposure periods. *Journal of chronic diseases*, 40:139S–161S, 1987.

Shpitser, I. and Pearl, J. Identification of joint interventional distributions in recursive semi-markovian causal models. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, pp. 1219. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006.

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

Tian, J. and Pearl, J. On the testable implications of causal models with hidden variables. In *Proceedings of the eighteenth conference on Uncertainty in Artificial Intelligence*, volume 1, pp. 519–527, 2002.

Tikka, S., Hyttinen, A., and Karvanen, J. Causal effect identification from multiple incomplete data sources: A general search-based approach. *arXiv preprint arXiv:1902.01073*, 2019.

Vitolo, C., Scutari, M., Ghalaieny, M., Tucker, A., and Russell, A. Modeling air pollution, climate, and health data using bayesian networks: A case study of the english regions. *Earth and Space Science*, 5(4):76–88, 2018.

This appendix is organized as follows.

- In Section A, we review necessary definitions and known results from the literature.
- In Section B, we provide the proofs for all of our results stated in the main text and the appendix.
- In Section C, we discuss various cases where the min-cost intervention problem can be solved more efficiently (i.e., we provide polynomial time algorithms) under assumptions on the structure of the causal graph, or the cost function.
- In Section D, we provide the details of our proposed heuristic algorithms.
- Section E includes further details on the approximation algorithm used to solve the hitting set algorithm, as well as a slight modification to Algorithm 2 to reduce the number of calls to solve the hitting set problem.
- Section F provides further details of the experimental setup of this paper, along with further evaluations of the proposed algorithms in this work.

A. Preliminaries

We begin with the definition of *structural causal model* (SCM) (Pearl et al., 2000), which is the framework we have used throughout this paper. An SCM is a tuple $M = (U, V, F, P(U))$, where U is the set of exogenous variables which are not observed but affect the relationship among the variables of the system, $V = \{v_1, ..., v_n\}$ is the set of observed endogenous variables where each $v_i \in V$ is a function of a subset of $V \cup U$ denoted by $\mathbf{pa}(v_i) \cup \mathbf{pa}^U(v_i)$, $F = \{f_1, ..., f_n\}$ is a set of functions where each f_i determines the value of $v_i = f_i(\mathbf{pa}(v_i), \mathbf{pa}^U(v_i))$, and $P(U)$ is the joint probability distribution over the variables U. An intervention is defined through a mathematical operator $do(X = \hat{X})$, which replaces the functions corresponding to variables X in the model M with a constant function $f = \hat{X}$. Denoting this model by $M_{\hat{X}}$, the interventional distribution $P(Y|do(X = \hat{X}))$ is then given by $P_{M_{\hat{X}}}(Y)$, or $P_{\hat{X}}(Y)$ in short (Pearl, 2012). For a subset S of variables V, we denote by $Q[S] = P(S|do(V \setminus S))$, the interventional distribution of the variables S after intervention on the rest of the variables.

The causal graph corresponding to the SCM M is a semi-Markovian graph $\mathcal{G}$ with one vertex for each $v_i \in V$, where there is a directed edge from v_i to v_j if the value of v_i is a function of v_j ($v_j \in \mathbf{pa}(v_i)$), and there is a bidirected edge between v_i and v_j if the values of v_i and v_j are both functions of a common exogenous variable u. We use V to denote the set of vertices of $\mathcal{G}$ throughout the paper. We use the words vertex and variable interchangeably throughout this work, as each vertex represents a variable. We utilise common graph-theoretic terms such as parents of a vertex x (denoted by $\mathbf{pa}(x)$), as well as children, ancestors, and descendants of a vertex. We denote by $\mathbf{biD}(x)$, the set of vertices that have a bidirected edge to x. We also denote by $\mathbf{pa}^{\leftrightarrow}(x) = \mathbf{pa}(x) \cap \mathbf{biD}(x)$ the set of parents of x that have a bidirected edge to x. For a set X, $\mathbf{pa}(X)$ is defined as $\mathbf{pa}(x) = \cup_{x \in X} \mathbf{pa}(x) \setminus X$. The rest of the aforementioned sets are defined analogously for a set of variables X. For a set of variables X, we denote by $\mathcal{G}_{[X]}$ the induced vertex subgraph of $\mathcal{G}$ over the vertices X. Following Pearl's notation, for two sets of variables X and Y, the graph $\mathcal{G}_{\overline{X}\underline{Y}}$ is defined as the edge subgraph of $\mathcal{G}$, where the edges going into X and the edges going out of Y are deleted.

Definition 8 (Causal identification)**.** *Let $\mathcal{G}$ be a semi-Markovian graph. We say the causal effect of the intervention $do(Y)$ on X (i.e., $P(X|do(Y))$) is identifiable in $\mathcal{G}$, if for any two positive models M_1 and M_2 that induce the causal graph $\mathcal{G}$, $P_{M_1}(V) = P_{M_2}(V)$ implies $P_{M_1}(X|do(Y)) = P_{M_2}(X|do(Y))$.*

A generalization of the definition above, which is used throughout this work is as follows. We say the causal effect of the intervention $do(Y)$ on X (i.e., $P(X|do(Y))$) is identifiable in $\mathcal{G}$ from the set of interventional distributions $\mathbf{P} = \{P(X_1|do(Y_1)), ..., P(X_k|do(Y_k))\}$, if for any two positive models M_1 and M_2 that induce the causal graph $\mathcal{G}$, $P_{M_1}(X_i|do(Y_i)) = P_{M_2}(X_i|do(Y_i))$ for $1 \leq i \leq k$ implies $P_{M_1}(X|do(Y)) = P_{M_2}(X|do(Y))$. Letting $\mathbf{P}$ be $\mathbf{P} = \{P(V|do(\emptyset))\}$, this generalization reduces to Definition 8. Kivva et al. (2022) prove that Pearl's do calculus is complete to test the identifiability of a causal effect from a set of distributions $\mathbf{P}$ under this definition.

The definitions of root set, c-component, c-forest and hedge are all adopted from (Shpitser & Pearl, 2006).

Definition 9 (C-component)**.** *Let $\mathcal{G}$ be a semi-Markovian graph. $\mathcal{G}$ is a c-component (confounded-component) if a subset of its bidirected edges form a spanning tree over all vertices of $\mathcal{G}$.*

Remark 4. *If $\mathcal{G}$ is not a c-component, it can be uniquely partitioned into maximal c-components.*

Definition 10 (Root set)**.** *We say R is a root set in $\mathcal{G}$ if for every $r \in R$, the set of descendants of r in $\mathcal{G}$ is empty.*

Definition 11 (C-forest)**.** *Let $\mathcal{G}$ be a semi-Markovian graph with the maximal root set R. $\mathcal{G}$ is a R-rooted c-forest if $\mathcal{G}$ is a c-component and all the observed variables have at most one child.*

Definition 12 (Hedge)**.** *Let X, Y be two set of vertices in the semi-Markovian graph $\mathcal{G}$. Also let F, F' be two R-rooted c-forests such that $F' \subset F$, $F \cap X \neq \emptyset$, $F' \cap X = \emptyset$ and R is a subset of ancestors of Y in $\mathcal{G}_{\overline{X}}$. Then F, F' form a hedge for $P(Y|do(X))$ in $\mathcal{G}$.*

Theorem 5 ((Shpitser & Pearl, 2006))**.** *If there exists a hedge formed for $P(Y|do(X))$ in $\mathcal{G}$, then $P(Y|do(X))$ is not identifiable in $\mathcal{G}$.*

Remark 5. *As mentioned in Theorem 6 of (Shpitser & Pearl, 2006), if an edge subgraph of $\mathcal{G}$ contains a hedge formed for $P(Y|do(X))$, then $P(Y|do(X))$ is not identifiable in $\mathcal{G}$. In other words, if $P(Y|do(X))$ is not identifiable in $\mathcal{G}$, it is not identifiable in any edge super-graph of $\mathcal{G}$ either.*

For the purposes of this paper where we are predominantly considering interventional distributions of the form $Q[S] = P(S|do(V \setminus S))$, we adapt the definition of hedge and the hedge criterion (Theorem 5) as follows. Let S be a subset of the vertices of $\mathcal{G}$ such that $\mathcal{G}_{[S]}$ is a c-component. First note that if F, F' form a hedge for $P(Y|do(X))$, then $(F \cup Y), (F' \cup Y)$ clearly form a hedge for $P(Y|do(X))$ by definition. Further, if the two R-rooted c-forests F, F' form a hedge for $Q[S]$, the set R must be S itself, as S has no other ancestors in $\mathcal{G}_{\overline{V \setminus S}}$ that can be a member of the root set. Consequently, $F' = R = S$, and F is a subset of $\mathcal{G}$ containing S. Also taking Remark 5 into consideration, a hedge formed for $Q[S]$ in $\mathcal{G}$ can be thought of as the following structure, which is the definition used throughout this paper.

Definition 4. *(Hedge) Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices such that $\mathcal{G}_{[S]}$ is a c-component. A subset F is a hedge formed for $Q[S]$ in $\mathcal{G}$ if $S \subsetneq F$, F is the set of ancestors of S in $\mathcal{G}_{[F]}$, and $\mathcal{G}_{[F]}$ is a c-component.*

Definitions 12 and 4 coincide when $\mathcal{G}_{[S]}$ is a c-component, and we used Definition 4 to simplify the text.

Claim 1. *A hedge w.r.t. Definition 4 is formed for $Q[S]$ if and only if a hedge w.r.t. Definition 12 is formed for $Q[S]$.*

Proof. Let F be a hedge formed for $Q[S]$ w.r.t. Definition 4. Taking $F' = S$, $X = V \setminus S$, $Y = S$, and $R = S$, the pair $F.F'$ forms a hedge for $Q[S] = P_X(Y)$ w.r.t. Definition 12. Conversely, if the pair F, F' is a hedge by definition 12, as argued above, $F' = R = S$. In this case, by definition of R-rooted c-forest, F is the set of ancestors of S in $\mathcal{G}_{[F]}$, and $\mathcal{G}_{[F]}$ is a c-component, which means that F forms a hedge for $Q[S]$ w.r.t. Defintion 4. $\square$

We also make use of the following theorem along our proofs.

Theorem 6 ((Tian & Pearl, 2002))**.** *Let $\mathcal{G}$ be a semi-Markovian graph, and let H be a subset of the observable vertices. Let $H_1, ..., H_k$ denote the maximal c-components of $\mathcal{G}_{[H]}$. Then $Q[H]$ is identifiable in $\mathcal{G}$, if and only if $Q[H_1], ..., Q[H_k]$ are identifiable in $\mathcal{G}$.*

A.1. Pearl's do Calculus

For the sake of completeness, we cite the three rules of Pearl's do calculus (Pearl, 2012).

Rule 1) Insertion or deletion of observations. If $(Y \perp\!\!\!\perp Z|X, W)_{\mathcal{G}_{\overline{X}}}$, then

$$P(Y|do(X), Z, W) = P(Y|do(X), W).$$

Rule 2) Action and observation exchange. If $(Y \perp\!\!\!\perp Z|X, W)_{\mathcal{G}_{\overline{X}\underline{Z}}}$, then

$$P(Y|do(X, Z), W) = P(Y|do(X), Z, W).$$

Rule 3) Insertion or deletion of actions. If $(Y \perp\!\!\!\perp Z|X, W)_{G_{\overline{XZ(W)}}}$, where $Z(W)$ are vertices in Z that have no descendants in W, then

$$P(Y|do(X, Z), W) = P(Y|do(X), W).$$

B. Proofs

Lemma 1. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a single c-component. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets of V such that $A_\cup \cap S = \emptyset$, where $A_\cup := \cup_{i=1}^m A_i$. If $\mathbf{A} \in \boldsymbol{ID}_\mathcal{G}(S, V \setminus S)$, then the singleton collection $\mathbf{A}_\cup = \{A_\cup\}$ also belongs to $\boldsymbol{ID}_\mathcal{G}(S, V \setminus S)$.*

Proof. Suppose $B_i = V \setminus A_i$ for $1 \leq i \leq m$, and define $B_\cap = \cap_{i=1}^k B_i$. Also suppose that $Q[S]$ is identifiable from the collection $\{Q[B_1], ..., Q[B_m]\}$. We claim that $Q[S]$ is also identifiable from $Q[B_\cap]$. Suppose not. Then there exists two structural equation models M_1 and M_2 on the set of variables V such that $Q^{M_1}[B_\cap] = Q^{M_2}[B_\cap]$, but $Q^{M_1}[S] \neq Q^{M_2}[S]$, where Q^{M_j} is the interventional distribution under model M_j.

Now we build two models M_1' and M_2' as follows. For any $x \in B_\cap$, x has the same equation in M_j' as in M_j. Both in M_1' and M_2', any $x \notin B_\cap$, x is uniformly distributed in $\mathcal{D}(x)$, where $\mathcal{D}(x)$ is the domain of the variable x. Since every $x \notin B_\cap$ is drawn independently of every other variable, for $1 \leq i \leq m$ we can write:

$$\begin{aligned}
Q^{M_1'}[B_i] &= Q^{M_1'}[B_\cap] \cdot \Pi_{x \in B_i \setminus B_\cap} Q^{M_1'}[x] \\
&= Q^{M_1}[B_\cap] \cdot \Pi_{x \in B_i \setminus B_\cap} \frac{1}{|\mathcal{D}(x)|} \\
&= Q^{M_2}[B_\cap] \cdot \Pi_{x \in B_i \setminus B_\cap} \frac{1}{|\mathcal{D}(x)|} \\
&= Q^{M_2'}[B_\cap] \cdot \Pi_{x \in B_i \setminus B_\cap} Q^{M_2'}[x] \\
&= Q^{M_2'}[B_i],
\end{aligned} \tag{4}$$

where the second equality follows from the fact that every variable in $B_\cap$ has the same model in M_1 and M_1', the third equality follows from the assumption that $Q^{M_1}[B_\cap] = Q^{M_2}[B_\cap]$, and the fourth one is because every variable in $B_\cap$ has the same model in M_2 and M_2'.

With the same line of reasoning as above,

$$Q^{M_1'}[S] = Q^{M_1}[S] \neq Q^{M_2}[S] = Q^{M_2'}[S]. \tag{5}$$

Equations (4) and (5) illustrate that $Q[S]$ is unidentifiable from the collection $\{Q[B_1], ..., Q[B_m]\}$, which contradicts the assumption of the lemma. Therefore, $Q[S]$ must be identifiable from $Q[B_\cap]$, or equivalently, $\mathbf{A} = \{A_\cup\} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S)$. □

Theorem 1. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets such that $\mathbf{A} \in \boldsymbol{ID}_\mathcal{G}(S, V \setminus S)$ and $m > 1$. Then, there exists a subset $\tilde{A} \subseteq V$ such that $\tilde{\mathbf{A}} = \{\tilde{A}\} \in \boldsymbol{ID}_\mathcal{G}(S, V \setminus S)$ and $\mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A})$.*

Proof. Suppose without loss of generality that $A_i \cap S = \emptyset$ for $1 \leq i \leq k$, and $A_i \cap S \neq \emptyset$ for $k < i \leq m$, for some integer k. We first claim that the following collection is in $\mathbf{ID}_\mathcal{G}(S, V \setminus S)$.

$$\hat{\mathbf{A}} = \{A_1, ..., A_k\} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S).$$

Suppose this claim does not hold. Then from theorem 1 of (Kivva et al., 2022), $Q[S]$ is not identifiable from any of $Q[A_i]$s for $1 \leq i \leq k$. Since for $i > k$, we have $S \not\subseteq A_i$, applying Theorem 1 of (Kivva et al., 2022) again, $\mathbf{A} \notin \mathbf{ID}_\mathcal{G}(S, V \setminus S)$, which is a contradiction.

Now defining $A_\cup = (\cup_{i=1}^k A_i)$, from Lemma 1, we know that

$$\tilde{\mathbf{A}} = \{A_\cup\} \in \mathbf{ID}_\mathcal{G}(S, V \setminus S).$$

It suffices to show that $\mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A})$, which follows from an identical reasoning to Remark 2:

$$\begin{aligned}
\mathbf{C}(\tilde{\mathbf{A}}) = \mathbf{C}(A_\cup) = \sum_{a \in A_\cup} \mathbf{C}(a) &\leq \sum_{i=1}^k \sum_{a \in A_i} \mathbf{C}(a) \\
&= \sum_{i=1}^k \mathbf{C}(A_i) = \mathbf{C}(\hat{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A}).
\end{aligned}$$

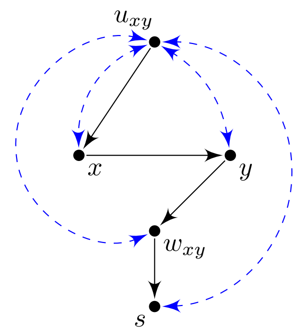


Figure 4. Reduction from the weighted vertex cover problem to the minimum-cost intervention problem. Each edge $\{x, y\}$ in the undirected graph $\mathcal{H}$ is represented by a hedge structure in the semi-Markovian graph $\mathcal{G}$.

□

Lemma 2. *Suppose S is a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. If $A \in \mathbf{ID}_1(S)$, then $A \cap S = \emptyset$.*

Proof. Suppose $A \cap S$ is nonempty, and $s \in A \cap S$ is an arbitrary variable. Define two models M_1 and M_2 as follows. Every variable in M_1 is uniformly drawn from $\{0, 1\}$. Also, every variable in M_2 except s is uniformly drawn from $\{0, 1\}$, and s is drawn from $\{0, 1\}$ with probabilities 0.4 and 0.6, respectively. Let Q^{M_i} denote the interventional distributions under model M_i, for $i \in \{1, 2\}$. Clearly, $Q^{M_1}[V \backslash A] = Q^{M_2}[V \backslash A] = \frac{1}{2^{|V|-|A|}}$, whereas $Q^{M_1}[S_{s=0}] = \frac{1}{2^{|S|}} \neq \frac{1}{0.4 * 2^{|S|-1}} = Q^{M_2}[S_{s=0}]$, which shows that $Q[S]$ is not identifiable in $\mathcal{G}_{[V \backslash A]}$.

□

Theorem 2. *WMVC problem is reducible to a min-cost intervention problem in polynomial time.*

Proof. Suppose an undirected graph $\mathcal{H} = (V_\mathcal{H}, E_\mathcal{H})$ along with a weight function $\omega : V_\mathcal{H} \to \mathbb{R}^{\geq 0}$ is given. We construct a semi-Markovian graph $\mathcal{G}$ along with a cost function $\mathbf{C}$ and prove that the min vertex cover problem in $\mathcal{H}$ is equivalent to the min-cost intervention problem in $\mathcal{G}$ for some set S. The construction is as follows.

We first begin with defining the vertex set of $\mathcal{G}$. For any vertex $x \in V_\mathcal{H}$, add a vertex x in $\mathcal{G}$. For any edge $\{x, y\} \in E_\mathcal{H}$, add two vertices u_{xy} and w_{xy} in $\mathcal{G}$. We will denote the set of all such vertices by U and W, respectively. Finally, add a vertex s. The number of vertices of $\mathcal{G}$ (denoted by $V = V_\mathcal{H} \cup U \cup W \cup \{s\}$) is therefore equal to $(|V_\mathcal{H}| + 2|E_\mathcal{H}| + 1)$. Assume a random ordering σ on the vertices of $\mathcal{H}$. Now take an edge $\{x, y\} \in E_\mathcal{H}$, and assume without loss of generality that x precedes y in σ. Add the directed edges $u_{xy} \to x, x \to y, y \to w_{xy}$, and $w_{xy} \to s$. Also draw a bidirected edge between u_{xy} and all of the vertices $\{x, y, w_{xy}, s\}$. Graph $\mathcal{G}$ has therefore $4|E_\mathcal{H}|$ directed and $4|E_\mathcal{H}|$ bidirected edges (4 edges for each edge in $\mathcal{H}$). Figure 4 demonstrates the structure corresponding to the the edge $\{x, y\}$ constructed in $\mathcal{G}$. Finally, the cost function $\mathbf{C}$ is defined as follows. For $x \in V_\mathcal{H}$, $\mathbf{C}(x) = \omega(x)$. for every other vertex $y \in U \cup W \cup \{s\}$, $\mathbf{C}(y) = z$, where

$$z = |V_\mathcal{H}| \cdot \max_{x \in V_\mathcal{H}} \omega(x) + 1.$$

First note that constructing the graph $\mathcal{G}$ and the cost function $\mathbf{C}$ given $\mathcal{H}$ and ω can be done in polynomial time, as it only needs a sweep over the vertices and the edges of $\mathcal{H}$, which can be performed in time $\mathcal{O}(V_\mathcal{H} + E_\mathcal{H})$. To complete the proof of the theorem, we will show that a subset $A \subseteq V_\mathcal{H}$ is a weighted minimal vertex cover for $\mathcal{H}$ if and only if A is a minimum-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$. We begin with the following claims.

Claim 1: $\{V_\mathcal{H}\} \in \mathbf{ID}_\mathcal{G}(\{s\})$. To see this, we simply provide the identification formula.

$$\begin{aligned} Q[\{s\}] &= P(s|do(V_\mathcal{H}, U, W)) \\ &= P(s|W, do(V_\mathcal{H}, U)) \quad \text{(do calculus rule 2)} \\ &= P(s|W, do(V_\mathcal{H})). \quad \text{(do calculus rule 3)} \end{aligned}$$

As seen in the expression above, $Q[\{s\}]$ can be identified by intervening on (fixing) only the variables $V_\mathcal{H}$.

Claim 2: if A is a minimum-cost intervention to identify $Q[\{s\}]$, then $A \subseteq V_{\mathcal{H}}$. First note that from claim 1, we know that the cost of the min-cost intervention is at most $\mathbf{C}(V_{\mathcal{H}}) \leq |V_{\mathcal{H}}| \cdot \max_{x \in V_{\mathcal{H}}} \mathbf{C}(x) \leq (z-1)$. Since the cost of every variable in $V \setminus V_{\mathcal{H}}$ is equal to z, the min-cost intervention clearly does not include any such variable.

Claim 3: if A is a min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$, then A is a vertex cover for $\mathcal{H}$. Take an arbitrary edge $\{x, y\} \in E_{\mathcal{H}}$. To prove this claim, it suffices to show that either $x \in A$ or $y \in A$. The structure $\mathcal{G}_{[x,y,u_{xy},w_{xy},s]}$ (as depicted in Figure 4) is a hedge formed for $Q[\{s\}]$ in $\mathcal{G}$. Since $\{A\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$, at least one of the variables $\{x, y, u_{xy}, w_{xy}\}$ is included in A, as otherwise the aforementioned hedge precludes the identification of $Q[\{s\}]$. However, from claim 2 we know that $A \subseteq V_{\mathcal{H}}$, and therefore at least one of x, y is in A, which completes the proof of the claim.

claim 4: if A is a vertex cover for $\mathcal{H}$, then $\{A\} \in \mathbf{ID}_{\mathcal{G}}(\{s\}, V \setminus \{s\})$ in $\mathcal{G}$. Again an identification formula based on the do-calculus rules can be derived. We first begin with the third rule of do calculus to derive $Q[\{s\}] = P(s|do(V_{\mathcal{H}}, U, W)) = P(s|do(A, U, W))$. This is based on the fact that $s \perp\!\!\!\perp (V \setminus A)|A, U, W$ in the graph where incoming edges to $U, W, V \setminus A$ are deleted. Now similar to the arguments of claim 1,

$$\begin{aligned} Q[\{s\}] &= P(s|do(V_{\mathcal{H}}, U, W)) = P(s|do(A, U, W)) \\ &= P(s|W, do(A, U)) \quad \text{(do calculus rule 2)} \\ &= P(s|W, do(A)). \qquad \text{(do calculus rule 3)} \end{aligned} \tag{6}$$

In the last equality, we used the fact that A is a vertex cover for $\mathcal{H}$, and therefore in every structure like the one shown in Figure 4, at least one of the vertices x or y is included in A, i.e., non of the vertices in U have a direct path to s in the graph where the incoming edges to A are deleted. Equation 6 proves claim 4, as intervention on A suffices to identify $Q[\{s\}]$.

Now suppose A is a minimum vertex cover for $\mathcal{H}$. From claim 4, $\{A\} \in \mathbf{ID}_{\mathcal{G}}(\{s\}, V \setminus \{s\})$. We claim that this intervention is a minimum-cost intervention to identify $Q[\{s\}]$. Suppose not. Then there exists a min-cost intervention $\hat{A}$ and $\mathbf{C}(\hat{A}) < \mathbf{C}(A)$. From claim 3, $\hat{A}$ is a vertex cover for $\mathcal{H}$. By definition of $\mathbf{C}(\cdot)$, $\sum_{a \in \hat{A}} \omega(a) = \mathbf{C}(\hat{A}) < \mathbf{C}(A) = \sum_{a \in A} \omega(a)$, which contradicts the assumption that A is the min vertex cover.

Conversely, suppose A is the min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$. From claim 3, A is also a vertex cover for $\mathcal{H}$. We claim that this vertex cover is a minimum vertex cover. Suppose not. Then there exists a minimum vertex cover $\hat{A}$ for $\mathcal{H}$ and $\sum_{a \in \hat{A}} \omega(a) < \sum_{a \in A} \omega(a)$. From claim 4, $\{\hat{A}\} \in \mathbf{ID}_{\mathcal{G}}(\{s\}, V \setminus \{s\})$. By definition of $\mathbf{C}(\cdot)$, $\mathbf{C}(\hat{A}) = \sum_{a \in \hat{A}} \omega(a) < \sum_{a \in A} \omega(a) = \mathbf{C}(A)$, which contradicts the assumption that A is the min-cost intervention. ☐

Remark 3. *The unweighted version of WMVC problem (i.e., when the weight function is given by $\omega(\cdot) = 1$) can be reduced to a minimum-cost intervention problem with the constant cost function $\mathbf{C}(\cdot) = 1$ in polynomial time.*

Proof. The proof is analogous to the proof of Theorem 2 with slight modifications as follows. The graph $\mathcal{G}$ is constructed exactly in the same manner, but the cost function is forced to the constant $\mathbf{C}(\cdot) = 1$. With the exact same arguments of the proof of Theorem 2, A is a minimum vertex cover for $\mathcal{H}$ only if it is a min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$. The other direction does not necessarily hold, as claim 2 of that proof does not hold anymore. However, we show how a min-cost intervention solution A can be turned into a minimum vertex cover for $\mathcal{H}$. Suppose A is a min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$. Substitute any vertex $u_{xy} \in A \cap U$ or $w_{xy} \in A \cap W$ with one of the vertices x, y arbitrarily, to form the set $\hat{A}$. First note that $\mathbf{C}(\hat{A}) \leq \mathbf{C}(A)$, since the cost of all variables are the same. Also $\hat{A} \subseteq V_{\mathcal{H}}$. We claim that $\hat{A}$ is a vertex cover for $\mathcal{H}$. Take an arbitrary edge $\{x, y\} \in E_{\mathcal{H}}$. It suffices to show that at least one of x, y is included in $\hat{A}$, and follows from the fact that at least one of the variables x, y, u_{xy}, w_{xy} must appear in A, since otherwise $\{x, y, u_{xy}, w_{xy}, s\}$ is a hedge formed for $Q[\{s\}]$ in G after intervention on A, which contradicts the fact that $\{A\} \in \mathbf{ID}_{\mathcal{G}}(\{s\}, V \setminus \{s\})$.

Note that $\hat{A}$ is also a minimum vertex cover for $\mathcal{H}$, since otherwise any vertex cover with smaller weight would also be an intervention with smaller cost than $\mathcal{A}$ to identify $Q[\{s\}]$, which is a contradiction. ☐

Lemma 3. *Let $\mathcal{G}$ be a semi-Markovian graph with vertex set V, along with a cost function $\mathbf{C} : V \to \mathbb{R}^{\geq 0}$. Let S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Suppose the set of all hedges formed for $Q[S]$ in $\mathcal{G}$ is $\{F_1, ..., F_m\}$. Then A^*_S is a solution to Equation* (2) *if and only if it is a solution to the MWHS problem for the sets $\{F_1 \setminus S, ..., F_m \setminus S\}$, with the weight function $\omega(\cdot) := \mathbf{C}(\cdot)$.*

Proof. First note that if A is an intervention set that makes $Q[S]$ identifiable, it hits all the hedges formed for $Q[S]$ in $\mathcal{G}$, as otherwise from hedge criterion (Theorem 5) $Q[S]$ would not be identifiable. Conversely, if A hits all the hedges formed for

$Q[S]$ in $\mathcal{G}$, intervening on A makes $Q[S]$ identifiable. As a result, a set A is an intervention to identify $Q[S]$ in $\mathcal{G}$ if and only if it is a hitting set for $\{F_1 \setminus S, ..., F_m \setminus S\}$. Since the set of interventions and the hitting sets coincide, a minimum intervention set is a minimum hitting set and vice-versa. □

Lemma 4. *Let $\mathcal{G}$ be a semi-Markovian graph with the vertex set V, and for $S \subseteq V$, let $\mathcal{G}_{[S]}$ be a c-component. For any subset $A \subseteq V$, if $A \in \mathbf{ID}_1(S)$, then $\mathbf{pa}^{\leftrightarrow}(S) \subseteq A$.*

Proof. First, from Lemma 2 we know that $S \cap A = \emptyset$. Now define $B = S \cup (\mathbf{pa}^{\leftrightarrow}(S) \setminus A)$. If $B \setminus S \neq \emptyset$, then by definition, B is a hedge formed for $Q[S]$ in $\mathcal{G}_{[V \setminus A]}$, and from Theorem 5, $Q[S]$ is not identifiable in $\mathcal{G}_{[V \setminus A]}$, which is a contradiction. As a result, $B \setminus S = \emptyset$, or equivalently, $\mathbf{pa}^{\leftrightarrow}(S) \setminus A = \emptyset$. □

Lemma 5. *Consider A^*_S in Equation (2), then $A^*_S \subseteq Hhull(S, \mathcal{G}) \setminus S$.*

Proof. By definition of hedge hull, all the hedges formed for $Q[S]$ are a subset of $Hhull(S)$. From Lemma 3, $A^* \subseteq Hhull(S)$. The result now follows from Lemma 2, which states that $A^* \cap S = \emptyset$. □

Lemma 6. *Given a semi-Markovian graph $\mathcal{G}$ over V and a subset $S \subseteq V$ such that $\mathcal{G}_{[S]}$ is a c-component, Algorithm 1 returns $Hhull(S, \mathcal{G})$ in $\mathcal{O}(|V|^3)$.*

Proof. First note that every time that F_2 is always a subset of F throughout the algorithm. Therefore, every time that $F_2 \neq F$, at least one vertex is excluded from F. Hence, the while loop is performed at most $|V|$ times, and the algorithm ends. Inside every loop, two depth first searches are executed, one to find the connected component of S in the edge-induced subgraph of $\mathcal{G}_{[F]}$ over its bidirected edges, and the other to find the ancestors of S in $\mathcal{G}_{[F]}$. DFS is quadratic-time in the worst case, i.e., each iteration runs in time $2|F|^2 \leq 2|V|^2$ in the worst case. Therefore, the algorithm ends in time $\mathcal{O}(|V|^3)$.

Let $\tilde{F}$ be the output of Algorithm 1. Since in the last iteration $\tilde{F} = F_1 = F_2$, $\tilde{F}$ is the set of ancestors of S in $\mathcal{G}_{[\tilde{F}]}$, and also $\mathcal{G}_{[\tilde{F}]}$ is a c-component. By definition, $\tilde{F}$ is a hedge formed for $Q[S]$ in $\mathcal{G}$, and therefore $\tilde{F} \subseteq Hhull(S, \mathcal{G})$. Now suppose F' is a hedge formed for $Q[S]$ in $\mathcal{G}$. It suffices to show that $F' \subseteq \tilde{F}$. At the beginning of the algorithm, $F = V$, that is, F' is included in F. At each iteration, since every vertex in F' has a bidirected path to S through only the vertices in F', it also has a bidirected path to S in every subgraph of $\mathcal{G}$ which includes F'. As a result, when constructing the connected component of S in line 3 of Algorithm 1, $F' \subseteq F_1$. Further, by definition of hedge, every vertex in F' has a directed path to S that goes through only vertices of F'. By the same argument, every vertex in F' is included in F_2 in line 4. Therefore, $F' \subseteq \tilde{F}$. □

Theorem 3. *Let S be a subset of variables such that $\mathcal{G}_{[S]}$ is a c-component. Then, A^*_S is a solution to (2) if and only if both $\mathbf{pa}^{\leftrightarrow}(S) \subseteq A^*_S$ and $A^*_S \setminus \mathbf{pa}^{\leftrightarrow}(S)$ is a min-cost intervention to identify $Q[S]$ in $\mathcal{G}_{[H]}$, where*

$$H := Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]}). \tag{3}$$

Proof. First note that the set of hedges formed for $Q[S]$ in $\mathcal{G}$ can be partitioned into hedges that intersect with $\mathbf{pa}^{\leftrightarrow}(S)$, and the hedges that do not intersect with $\mathbf{pa}^{\leftrightarrow}(S)$, denoted by $\mathbf{F_1}$ and $\mathbf{F_2}$, respectively. The set of hedges formed for $Q[S]$ in $\mathcal{G}_{[H']}$ is then $\mathbf{F_2}$. Using Lemma 3, the lemma is equivalent to the claim that A^* is a minimum hitting set for the hedges $\mathbf{F_1} \cup \mathbf{F_2}$ if and only if $A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)$ is a minimum hitting set for hedges $\mathbf{F_2}$.

Suppose A^* is a min-cost intervention to identify $Q[S]$ in $\mathcal{G}$. From Lemma 3, A^* hits all the hedges formed for $Q[S]$ in $\mathcal{G}_{H'}$, i.e., $\mathbf{F_2}$. However, since none of these hedges intersect with $\mathbf{pa}^{\leftrightarrow}(S)$, $A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)$ hits all of these hedges. We claim that $A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)$ is the minimum hitting set for the hedges $\mathbf{F_2}$. Suppose there exists another set $\tilde{A}$ such that $\tilde{A}$ hits all the hedges $\mathbf{F_2}$, and $\mathbf{C}(\tilde{A}) < \mathbf{C}(A^* \setminus \mathbf{pa}^{\leftrightarrow}(S))$. Since all the hedges $\mathbf{F_1}$ intersect with $\mathbf{pa}^{\leftrightarrow}(S)$, $\tilde{A} \cup \mathbf{pa}^{\leftrightarrow}(S)$ hits all the hedges formed for $Q[S]$ in $\mathcal{G}$, and

$$\begin{aligned}
\mathbf{C}(\tilde{A} \cup \mathbf{pa}^{\leftrightarrow}(S)) &\leq \mathbf{C}(\tilde{A}) + \mathbf{C}(\mathbf{pa}^{\leftrightarrow}(S)) \\
&< \mathbf{C}(A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)) + \mathbf{C}(\mathbf{pa}^{\leftrightarrow}(S)) \\
&= \mathbf{C}(A^*),
\end{aligned}$$

which contradicts the fact that A^* is the minimum-cost intervention to identify $Q[S]$ in $\mathcal{G}$.

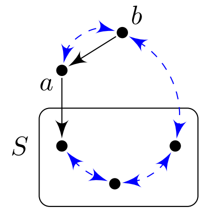


Figure 5. Hedges of size 2 must be in the form depicted above. Exactly one vertex a is a member of $\mathbf{pa}(S) \setminus \mathbf{biD}(S)$, and the other vertex b is a member of $\mathbf{biD}(S) \setminus \mathbf{pa}(S)$.

Conversely, let $A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)$ be a minimum hitting set for hedges $\mathbf{F_2}$. If A^* is not the min-cost intervention to identify $Q[S]$ in $\mathcal{G}$, then there exists $\tilde{A}$ such that $\mathbf{C}(\tilde{A}) < \mathbf{C}(A^*)$ and $\tilde{A}$ hits the hedges $\mathbf{F_1} \cup \mathbf{F_2}$. From Lemma 4, $\mathbf{pa}^{\leftrightarrow}(S) \subseteq \tilde{A}$. Since hedges $\mathbf{F_2}$ do not intersect with $\mathbf{pa}^{\leftrightarrow}(S)$, $\tilde{A} \setminus \mathbf{pa}^{\leftrightarrow}(S)$ hits all the hedges $\mathbf{F_2}$, and

$$\begin{aligned}\mathbf{C}(\tilde{A} \setminus \mathbf{pa}^{\leftrightarrow}(S)) &= \mathbf{C}(\tilde{A}) - \mathbf{C}(\mathbf{pa}^{\leftrightarrow}(S)) \\ &< \mathbf{C}(A^*) - \mathbf{C}(\mathbf{pa}^{\leftrightarrow}(S)) \\ &= \mathbf{C}(A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)),\end{aligned}$$

which contradicts the fact that $A^* \setminus \mathbf{pa}^{\leftrightarrow}(S)$ is the minimum-cost intervention to identify $Q[S]$ in $\mathcal{G}_{[H']}$. ☐

Lemma 7. *Let $\mathcal{G}$ be a semi-Markovian graph and $S \subseteq V$. Algorithm 2 returns an optimal solution to* (2)*.*

Proof. First let $\mathcal{G}_{[S]}$ be a c-component. Note that every time that the set A is constructed in line 12 and intervening on $A \cup \mathbf{pa}^{\leftrightarrow}(S)$ does not suffice to identify $Q[S]$ in $\mathcal{G}$, the algorithm continues on the subgraph of $\mathcal{G}$ over $Hhull(S, \mathcal{G}_{V\setminus(A\cup\mathbf{pa}^{\leftrightarrow}(S))})$. As a result, the newly discovered hedges do not intersect with A, i.e., Algorithm 2 never discovers redundant hedges. Since every hedge is a subset of the graph and the number of such subsets is finite, the algorithm halts within finite time. At each iteration (while loop of lines 5-11), a new hedge is added to the set of hedges formed for $Q[S]$ and adds it to the set $\mathbf{F}$. From the minimum hitting set formulation, it is clear that any min-cost intervention for identifying $Q[S]$ in $\mathcal{G}$ must intersect with all the hedges in $\mathbf{F}$. As a result, the output of Algorithm 2 (A) is always a subset of a min-cost intervention. Further, by construction $A \in \mathbf{ID}_1(S)$. As a result, A is a min-cost intervention for identifying $Q[S]$ in $\mathcal{G}$.

Now suppose $\mathcal{G}_{[S]}$ is not a c-component. $\mathcal{G}_{[S]}$ can be uniquely partitioned into its maximal c-components $\mathcal{G}_{[S_1]}, ..., .\mathcal{G}_{[S_k]}$. The arguments above hold for any of the maximal c-components $\mathcal{G}_{[S_i]}$. The result follows from the fact that $Q[S]$ is identifiable in $\mathcal{G}$ if and only if all of its maximal c-components are identifiable in $\mathcal{G}$. ☐

Lemma 8. *Let $\mathcal{G}$ be a semi-Markovian graph on V and S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Heuristic Algorithm 1 returns an intervention set A in $\mathcal{O}(|V|^3)$ such that $A \in \mathbf{ID}_1(S)$.*

Proof. *Correctness.* Let the output of the algorithm be A. We will utilize the hitting set formulation to show the correctness of the algorithm. Let F be a hedge formed for $Q[S]$ in $\mathcal{G}$. It suffices to show that $F \cap A \neq \emptyset$. If $F \cap \mathbf{pa}^{\leftrightarrow}(S) \neq \emptyset$, then the claim holds since $\mathbf{pa}^{\leftrightarrow}(S) \subseteq A$. Otherwise, F is a hedge formed for $Q[S]$ in $\mathcal{G}_{[V\setminus\mathbf{pa}^{\leftrightarrow}(S))]}$, i.e., $F \subseteq H$, where H is given by Equation (3). Now let a be an arbitrary vertex in $(F \setminus S) \cap \mathbf{pa}(S)$. Such a vertex exists by definition of hedge. Further, $\mathcal{G}_{[F]}$ is a c-component, i.e., there exists a path from a to S through bidirected edges in F. As a result, in the undirected graph $\mathcal{H}$ built in heuristic Algorithm 1, there exists a path from x to y that passes only through vertices in F. Any solution to minimum vertex cut for $x - y$ must include at least one vertex of F. Therefore, $F \cap A \neq \emptyset$.

Runtime. Heuristic Algorithm 1 begins with constructing the set $\mathbf{pa}^{\leftrightarrow}(S)$, and the set H given by Equation (3), which are performed in time $\mathcal{O}(|V|)$, and $\mathcal{O}(|V|^3)$ in the worst case. Constructing the graph $\mathcal{H}$ requires iterating over the bidirected edges of $\mathcal{G}_{[H]}$, which can be done in time $\mathcal{O}(|H|^2)$ in the worst case. The reduction from minimum vertex cut to minimum edge cut discussed in Appendix D is linear-time. The final step of the algorithm is to solve a minimum edge cut, which can be done in time $\mathcal{O}(|H|^3)$ using the push-relabel algorithm to solve the equivalent maximum flow problem (Goldberg & Tarjan, 1988). Noting that $|H| \leq |V|$, the runtime of the algorithm is $\mathcal{O}(|V|^3)$.

☐

Lemma 9. *Let $\mathcal{G}$ be a semi-Markovian graph on V and S be a subset of V such that $\mathcal{G}_{[S]}$ is a c-component. Heuristic Algorithm 2 returns an intervention set A in $\mathcal{O}(|V|^3)$ such that $A \in \mathbf{ID}_1(S)$.*

Proof. Correctness. Let the output of the algorithm be A. We will utilize the hitting set formulation to show the correctness of the algorithm. Let F be a hedge formed for $Q[S]$ in $\mathcal{G}$. It suffices to show that $F \cap A \neq \emptyset$. If $F \cap \mathbf{pa}^{\leftrightarrow}(S) \neq \emptyset$, then the claim holds since $\mathbf{pa}^{\leftrightarrow}(S) \subseteq A$. Otherwise, F is a hedge formed for $Q[S]$ in $\mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]}$, i.e., $F \subseteq H$, where H is given by Equation (3). Now let b be an arbitrary vertex in $(F \setminus S) \cap \mathbf{biD}(S)$. Such a vertex exists by definition of hedge. Further, F are the ancestors of S in $\mathcal{G}_{[F]}$, i.e., there exists a directed path from b to S through directed edges in $\mathcal{G}_{[F]}$. As a result, in the directed graph $\mathcal{J}$ built in heuristic Algorithm 2, there exists a path from x to y that passes only through vertices in F. Any solution to minimum vertex cut for $x - y$ must include at least one vertex of F. Therefore, $F \cap A \neq \emptyset$.

Runtime. Heuristic Algorithm 2 begins with constructing the set $\mathbf{pa}^{\leftrightarrow}(S)$, and the set H given by Equation (3), which are performed in time $\mathcal{O}(|V|)$, and $\mathcal{O}(|V|^3)$ in the worst case. Constructing the graph $\mathcal{J}$ requires iterating over the directed edges of $\mathcal{G}_{[H]}$, which can be done in time $\mathcal{O}(|H|^2)$ in the worst case. The reduction from minimum vertex cut to minimum edge cut discussed in Appendix D is linear-time. The final step of the algorithm is to solve a minimum edge cut, which can be done in time $\mathcal{O}(|H|^3)$ using the push-relabel algorithm to solve the equivalent maximum flow problem (Goldberg & Tarjan, 1988). Noting that $|H| \leq |V|$, the runtime of the algorithm is $\mathcal{O}(|V|^3)$. □

Theorem 4. *Suppose S is a subset of variables such that $\mathbf{C}(s) = \infty$ for any $s \in S$. Let $\mathbf{A} = \{A_1, A_2, ..., A_m\}$ be a collection of subsets such that $\mathbf{A} \in \boldsymbol{ID}_{\mathcal{G}}(S, V \setminus S)$ and $m > 1$. Then there exists a singleton intervention $\tilde{A}$ such that $\tilde{\mathbf{A}} = \{\tilde{A}\} \in \boldsymbol{ID}_{\mathcal{G}}(S, V \setminus S)$ and $\mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A})$.*

Proof. First note that if there exists $1 \leq i \leq m$ such that $A_i \cap S \neq \emptyset$, then $\mathbf{C}(\mathbf{A}) = \infty$. In this case, $\tilde{A} = V \setminus S$ satisfies the desired property. Otherwise, we can assume that $S \cap (\cup_{i=1}^m A_i) = \emptyset$. We claim that $\tilde{A} = A_\cup = \cup_{i=1}^m A_i$ is the desired intervention set. To prove this claim, first note that

$$\mathbf{C}(\{\tilde{A}\}) = \mathbf{C}(A_\cup) = \sum_{a \in A_\cup} \mathbf{C}(a) \leq \sum_{i=1}^m \sum_{a \in A_i} \mathbf{C}(a)$$
$$= \sum_{i=1}^m \mathbf{C}(A_i) = \mathbf{C}(\mathbf{A}).$$

Therefore, it suffices to show that $\{\tilde{A}\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$. Let $S_1, ..., S_k$ denote the maximal c-components of $\mathcal{G}_{[S]}$. From lemma 2 of (Tian & Pearl, 2002) (restated here as Theorem 6), $Q[S]$ is identifiable in $\mathcal{G}_{V \setminus \tilde{A}}$, if and only if $Q[S_1], ..., Q[S_k]$ are identifiable in $\mathcal{G}_{V \setminus \tilde{A}}$. Therefore, it suffices to show that $\{\tilde{A}\} \in \mathbf{ID}_{\mathcal{G}}(S_i, V \setminus S_i)$, for all $1 \leq i \leq k$. This result follows from Lemma 1, since $\mathcal{G}_{[S_i]}$ is a c-component, $\{A_1, ..., A_m\} \in \mathbf{ID}_{\mathcal{G}}(S_i, V \setminus S_i)$ (from Theorem 6), and $A_\cup \cap S_i = \emptyset$. □

Lemma 10. *Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices. Suppose $\mathbf{A}^*_{S,V \setminus S}$ is a min-cost interventions collection to identify $Q[S]$ in $\mathcal{G}$. If $\mathcal{G}_{[S_j]}$ is a maximal c-component of $\mathcal{G}_{[S]}$, then there exists $A \in \mathbf{A}^*_{S,V \setminus S}$ such that $A \in \mathbf{ID_1}(S_j)$.*

Proof. Disclaimer: The proof of this result at initial submission (which is included below) was based on (Lee et al., 2020). Kivva et al. (2022) proved that the results of (Lee et al., 2020) are correct, although their proofs are incomplete. It is worth mentioning that this lemma immediately follows from Theorem 1 of (Kivva et al., 2022).

*Former proof.*With the same argument used in the proof of Theorem 1, suppose $\mathbf{A}^*_S = \{A_1, ..., A_m\}$ and suppose without loss of generality that $A_i \cap S_j = \emptyset$ for $1 \leq i \leq k$, and $A_i \cap S_j \neq \emptyset$ for $k < i \leq m$, for some integer k. We first claim that the following collection is in $\mathbf{ID}_{\mathcal{G}}(S_j, V \setminus S_j)$.

$$\hat{\mathbf{A}} = \{A_1, ..., A_k\} \in \mathbf{ID}_{\mathcal{G}}(S_j, V \setminus S_j). \tag{7}$$

Suppose this claim does not hold. Then from theorem 3 of (Lee et al., 2020), there exists a thicket $\mathcal{J}$ formed for $Q[S_j]$ with respect to $\hat{\mathbf{A}}$. By definition of thicket (definition 6 of (Lee et al., 2020)), $\mathcal{J}$ is also a thicket formed for $\mathbf{A}$, and from theorem 1 of (Lee et al., 2020), $Q[S_j]$ is not g-identifiable, or equivalently, $\mathbf{A} \notin \mathbf{ID}_{\mathcal{G}}(S_j, V \setminus S_j)$, which is a contradiction.

Now suppose there exists no $1 \leq i \leq k$ such that $A_i \in \mathbf{ID}_{\mathcal{G}}(S_j, V \setminus S_j)$. From Theorem 5, there exists a hedge formed for $Q[S_j]$ in $\mathcal{G}_{[V\setminus A_i]}$ for every $1 \leq i \leq k$. The set of all these k hedges forms a thicket for $Q[S_j]$ with respect to $\hat{\mathbf{A}}$, which is a contradiction to Equation (7) through theorem 1 of (Lee et al., 2020). □

Proposition 1. *Given a semi-Markovian graph $\mathcal{G}$ and a subset S of its vertices, Algorithm 3 with Algorithm 2 used as a subroutine in line (6) returns an optimal solution to the min-cost interventions collection to identify $Q[S]$ in $\mathcal{G}$.*

Proof. Let $\mathbf{A}$ be the output of Algorithm 3. We first claim that $\mathbf{A} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$. For any maximal c-component of $\mathcal{G}_{[S]}$ such as $\mathcal{G}_{[S_i]}$, there exists at least one set $A_j \in \mathbf{A}$ such that $A_j \in \mathbf{ID}_1(S)$. Therefore, $\mathbf{A} \in \mathbf{ID}_{\mathcal{G}}(S_i, V \setminus S_i)$. Since $Q[S_i]$ is identifiable from interventions on $\mathbf{A}$ for any c-component of $\mathcal{G}_{[S]}$, by the result form (Tian & Pearl, 2002), $Q[S]$ is also identifiable, i.e., $\mathbf{A} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$.

Now suppose $\mathbf{A}^*_S = \{A^*_1, ..., A^*_t\}$ is a min-cost intervention collection to identify $Q[S]$ in $\mathcal{G}$. It suffices to show that $\mathbf{C}(\mathbf{A}) \leq \mathbf{C}(\mathbf{A}^*_S)$. Consider an arbitrary partitioning of the c-components of S to t parts such as $S^{(1)}, ...S^{(t)}$, such that $A^*_i \in \mathbf{ID}_1(\underline{S}^{(i)})$ for $1 \leq i \leq t$. Note that such a partitioning is possible due to Lemma 10. Algorithm 3 considers this partitioning in one of its iterations, and due to optimality of Algorithm 2 (see Lemma 7), it constructs an intervention collection $\tilde{\mathbf{A}} = \{\tilde{A}_1, ..., \tilde{A}_t\}$ where $\tilde{A}_i$ is the optimal intervention set in $\mathbf{ID}_1(\underline{S}^{(i)})$ for $1 \leq i \leq t$. As a result,

$$\mathbf{C}(\tilde{\mathbf{A}}) = \sum_{i=1}^{t} mathbfC(TildeA_i) \leq \sum_{i=1}^{t} \mathbf{C}(A^*_i) = \mathbf{C}(\mathbf{A}^*_S).$$

Since Algorithm 3 outputs the minimum cost collection among all constructed intervention collections, clearly $\mathbf{C}(\mathbf{A}) \leq \mathbf{C}(\tilde{\mathbf{A}}) \leq \mathbf{C}(\mathbf{A}^*_S)$. □

Lemma 11. *Let $\mathcal{G}$ be a semi-Markovian graph such that the edge induced subgraphs of $\mathcal{G}$ over its directed edges and over its bidirected edges are trees. For an arbitrary vertex s and a vertex $x \in Hhull(s) \setminus \{s\}$, $NC_s(x)$ is a hedge formed for $Q[\{s\}]$ in $\mathcal{G}$.*

Proof. It suffices to show that $\mathcal{G}_{[NC_s(x)]}$ is a c-component and every vertex in $NC_s(x)$ is an ancestor of s in $\mathcal{G}_{[NC_s(x)]}$. Take an arbitrary vertex $y \in NC_s(x)$. By definition of $NC_s(\cdot)$, $nec_s(y) \subseteq NC_s(x)$. As a result, y has both a directed and a bidirected path to s in $\mathcal{G}_{[NC_s(x)]}$, i.e., y is an ancestor of s in $\mathcal{G}_{[NC_s(x)]}$, and in the same c-component as s in $\mathcal{G}_{[NC_s(x)]}$. Repeating the same argument for every vertex in $NC_s(x)$ completes the proof. □

Lemma 12. *Let $\mathcal{G}$ be a semi-Markovian graph over V such that the edge induced subgraphs of $\mathcal{G}$ over its directed edges and over its bidirected edges are trees. For a vertex s in $\mathcal{G}$, Algorithm 4 returns the min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$ in time $\mathcal{O}(|V|^3)$.*

Proof. First note that from Lemma 11, all the sets in $\mathbf{F}$ in line 9 of the algorithm are hedges formed for $Q[\{s\}]$ in $\mathcal{G}$. Therefore, any intervention to identify $Q[\{s\}]$ must hit all of these sets. On the other hand, if all of these sets are hit, by definition of $NC_s(\cdot)$, a hedge formed for $Q[\{s\}]$ cannot include any of the vertices in $Hhull(s)$, i.e., there does not exist any hedge formed for $Q[\{s\}]$ in $\mathcal{G}_{[V\setminus A]}$, where A is a hitting set for $\mathbf{F}$. As a result, the solution to min-cost intervention problem is the solution to minimum hitting set for $\mathbf{F}$ in line 9 of the algorithm. Further, we can eliminate any hedge F' from $\mathbf{F}$, if there is a hedge $F \in \mathbf{F}$ such that $F \subseteq F'$. This is due to the fact that if F is hit, F' is also hit. Observing that if $x \in NC_s(y)$ then $NC_s(x) \subseteq NC_s(y)$, we can eliminate all such sets $NC_s(y)$ from $\mathbf{F}$. At the end of this process (after the for loop of lines 10-13), we claim that for any two sets F, F' remaining in $\mathbf{F}$, $F \cap F' = \{s\}$. Suppose not. Then there exists $x \neq s$ such that $x \in F \cap F'$. Since F and F' are both sets of the Form $NC_s(\cdot)$, as mentioned above, $NC_s(x) \subseteq F$ and $NC_s(x) \subseteq F'$. Now if $NC_s(x) \in \mathbf{F}$, then both sets F and F' (or at least one of them, in the case that one of them is $NC_s(x)$ itself) would be eliminated from $\mathbf{F}$ during the for loop of lines 12-13. Otherwise, $NC_s(x) \notin \mathbf{F}$, which means that there exists a vertex $y \in H$ such that $NC_s(y) \subseteq NC_s(x)$, and therefore $NC_s(x)$ has been removed from $\mathbf{F}$. But in this case, $NC_s(y) \subseteq NC_s(x) \subseteq F, F'$, and in the same iteration where $NC_s(x)$ was removed from $\mathbf{F}$, both F and F' would also be eliminated. The contradiction shows that when the algorithm reaches line 14, all the sets $\{F \setminus \{s\} | F \in \mathbf{F}\}$ are disjoint. Clearly, the minimum hitting set for disjoint sets includes the minimum-cost member of each of the sets, which is the output of Algorithm 4, along with $\mathbf{pa}^{\leftrightarrow}(s)$ (Lemma 4).

As discussed earlier, constructing the hedge hull of s ($Hhull(s)$) has a worst-case time complexity of $\mathcal{O}(|V|^3)$. $\mathbf{pa}^{\leftrightarrow}(s)$ can also be constructed in linear time, using two one-step breadth-first searches. Let H denote the hedge hull of s in the graph $\mathcal{G}_{V\setminus\mathbf{pa}^{\leftrightarrow}(s)}$. Constructing $nec_s(x)$ for the vertices in H can be performed by solving two all-pair shortest paths, which requires two breadth-first search from each vertex with time complexity $\mathbf{O}(|H|^3)$ in the worst case. The while loop of lines 6-8 is performed at most $|H|$ times, each with linear complexity. As a result, the NC_s sets and therefore $\mathbf{F}$ are also constructed in time $\mathcal{O}(|H|^3)$. The for loop of lines 12-13 requires a sweep over the sets in $\mathbf{F}$, which are at most $|H|$ many sets, each with at most $|H|$ members; which can be performed in time $\mathcal{O}(|H|^2)$, and therefore the for loop of lines 10-13 has complexity $\mathcal{O}(|H|^3$. Finally, calculating the minimum of a set with at most $|H|$ members can be done in (sub)linear time. Therefore, the complexity of Algorithm 4 is $\mathbf{O}(|V|^3)$ in the worst case. ☐

Lemma 13. *Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices such that $\mathcal{G}_{[S]}$ is a c-component. Construct an undirected graph $\mathcal{H}$ on the same set of vertices as $\mathcal{G}\setminus \textbf{pa}^{\leftrightarrow}(S)$ as follows. For any hedge of size 2 formed for $Q[S]$ such as F, connect the two vertices in $F\setminus S$ with an edge. The resulting graph $\mathcal{H}$ is bipartite.*

Proof. First, let $F=\{a,b\}\cup S$ be a hedge formed for $Q[S]$ in the graph $\mathcal{G}\setminus\mathbf{pa}^{\leftrightarrow}(S)$. By definition of hedge, both vertices a,b are ancestors of S in $\mathcal{G}_{[F]}$. Therefore, at least one of these vertices must be a parent of S. Without loss of generality, assume $a\in\mathbf{pa}(S)$. Since $a\notin\mathbf{pa}^{\leftrightarrow}(S)$, a does not have a bidirected edge to any vertex in S. However, by definition of hedge, F is a c-component. Therefore, a must have a bidirected edge to b, and b has a bidirected edge to S. Further, $b\notin\mathbf{pa}^{\leftrightarrow}(S)$, and therefore $b\notin\mathbf{pa}(S)$. Since b must be an ancestor of S in $\mathcal{G}_{[F]}$, $b\in\mathbf{pa}(a)$. As a result, all hedges of size 2 are in the form depicted in Figure 5, where $a\in\mathbf{pa}(S)\setminus\mathbf{biD}(S)$ and $b\in\mathbf{biD}(S)\setminus\mathbf{pa}(S)$. Accordingly, for any edge drawn in $\mathcal{H}$ such as $\{a,b\}$, exactly one of them is in $\mathbf{biD}(S)\setminus\mathbf{pa}(S)$, and the other one is in $\mathbf{pa}(S)\setminus\mathbf{biD}(S)$. Partitioning the vertices of $\mathcal{H}$ into the aforementioned sets, it is clear that $\mathcal{H}$ is bipartite. ☐

Lemma 15. *Given a semi-Markovian graph $\mathcal{G}$ on V and a subset of its vertices S such that $\mathcal{G}_{[S]}$ is a c-component, Algorithm 9 returns a set A such that $\{A\}\in\mathbf{ID}_{\mathcal{G}}(S,V\setminus S)$ in time $\mathcal{O}(|V|^5)$ in the worst case.*

Proof. By construction, Algorithm 9 outputs a set A such that there is no hedge formed for $Q[S]$ in $\mathcal{G}_{[V\setminus A]}$. As a result, $\{A\}\in\mathbf{ID}_{\mathcal{G}}(S,V\setminus S)$. It only suffices to show that the algorithm halts in time $\mathcal{O}(|V|^5)$. Constructing the hedge hull in line 1 is performed in cubic time in the worst case. The while loop of lines 3-12 can only be executed $|H|$ times in the worst case, as at each iteration at least one vertex is removed from H. At each iteration of this loop, at most $|H|$ hedge hulls are constructed, where each of these operations can be done in time $\mathcal{O}(|H|^3)$. Summing these up, the algorithm runs in time $\mathcal{O}(|V|^3+|Hhull(S,\mathcal{G})|^5)$. ☐

Lemma 16. *Given a semi-Markovian graph on V and a subset S of its vertices, Algorithms 7, 8 and 9 return a subset A of the vertices of $\mathcal{G}$ such that $\{A\}\in\mathbf{ID}_{\mathcal{G}}(S,V\setminus S)$, in time $\mathcal{O}(|V|^3)$, $\mathcal{O}(|V|^3)$ and $\mathcal{O}(|V|^5)$, respectively.*

Proof. It is straightforward that the arguments used to prove the correctness of these algorithms for the case where $\mathcal{G}_{[S]}$ is a c-component still hold for any maximal c-component of $\mathcal{G}_{[S]}$ for an arbitrary subset S (see the proofs of Lemmas 8, 9 and 15.) Also, $Q[S]$ is identifiable in $\mathcal{G}$ if and only if all of its maximal c-components are identifiable (Tian & Pearl, 2002). The result follows immediately. It is worthy to note that the only overhead in the case that $\mathcal{G}_{[S]}$ is not a c-component is to partition S into its c-components, which can be done using DFS in time $\mathcal{O}(|V|^3)$ in the worst case, i.e., it does not alter the computational complexity of any of the heuristic algorithms. ☐

C. Special Cases & Improvements

In this section, we discuss a few special cases of the min-cost intervention problem, and how these cases can be solved efficiently. We show that under the assumption that the expert has certain knowledge about the structure of the causal graph $\mathcal{G}$, or the cost function $\mathbf{C}(\cdot)$, the problem of designing the minimum-cost intervention can be solved efficiently in polynomial time. Some of these assumptions might seem restrictive. However, as we shall discuss, they provide useful insight towards solving the min-cost intervention problem efficiently in more practical settings.

C.1. Tree-like structure of $\mathcal{G}$

We begin with a special structure of the semi-Markovian graph $\mathcal{G}$, where both the edge induced subgraphs of $\mathcal{G}$ over the directed edges and over the bidirected edges are trees. Between any pair of vertices in a tree, there is a unique path. As a result, for any two vertices a,b in $\mathcal{G}$, there is a unique path using bidirected edges, and if a is an ancestor of b, there is also a unique path from a to b using directed edges. We denote these unique bidirected and directed paths by $a\overset{p_u}{\longleftrightarrow}b$ and

Algorithm 4 Polynomial time algorithm for tree-like structures.

$H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]})$
$nec_s(x) \leftarrow (a \overset{p_u}{\longleftrightarrow} b \cup a \xrightarrow{p_u} b)$ for every $x \in H$
$NC_s(x) \leftarrow \{x\}$ for every $x \in H$
for $x \in H$ **do**
  $updated(y) \leftarrow false$ for every $y \in H$
  while $\exists y \in NC_s(x)$ s.t. $updated(y) = false$ **do**
    $NC_s(x) \leftarrow NC_s(x) \cup nec_s(y)$
    $updated(y) \leftarrow true$
$\mathbf{F} \leftarrow \{NC_s(x) | x \in H\}$
for $x \in H$ **do**
  if $NC_s(x) \in \mathbf{F}$ **then**
    for $y \in Hhull(s)$ s.t. $x \in NC_s(y)$ **do**
      $\mathbf{F} \leftarrow \mathbf{F} \setminus \{NC_s(y)\}$
$A \leftarrow \{\arg\min_{x \in F \setminus \{s\}} \mathbf{C}(x) | F \in \mathbf{F}\}$
return $A \cup \mathbf{pa}^{\leftrightarrow}(s)$

$a \xrightarrow{p_u} b$, respectively. Note that $a \overset{p_u}{\longleftrightarrow} b$ and $a \xrightarrow{p_u} b$ for every pair of vertices can be found using an all-pair shortest path algorithm (two separate breadth-first search from each vertex) in time $\mathcal{O}(|V|^3)$. Now suppose we want to solve the min-cost intervention set problem for $Q[\{s\}]$. Take an arbitrary variable $x \neq s$ from $Hhull(s)$. Let F be a hedge formed for $Q[\{s\}]$ such that $x \in F$. Since F is a c-component and x is an ancestor of s in $\mathcal{G}_{[F]}$, all of the variables on both $a \overset{p_u}{\longleftrightarrow} b$ and $a \xrightarrow{p_u} b$ must be members of F. We therefore call the union of all these variables, the necessary set of x to form a hedge for s, and we denote this set by $nec_s(x)$. Clearly, if we intervene on at least one vertex from $nec_s(x)$, then no hedge formed for $Q[\{s\}]$ contains x. Further, we observe that if $y \in nec_s(x)$, then with the same arguments, if a hedge formed for $Q[\{s\}]$ contains x, it must contain y and therefore all the variables in $nec_s(y)$ as well. We define the closure of necessary variables for x to form a hedge for $Q[\{s\}]$ as follows.

Definition 13 (Necessary closure)**.** *Let $\mathcal{G}$ be a semi-Markovian graph such that the edge induced subgraphs of $\mathcal{G}$ over its directed edges and over its bidirected edges are trees. We say a subset A of vertices of $\mathcal{G}$ is a closure of necessary variables for x to form a hedge for $Q[\{s\}]$, if $x \in A$, and for every $y \in A$, $nec_s(y) \subseteq A$. We denote the minimum closure of necessary variables for x by $NC_s(x)$.*

The following lemma indicates that minimum closure of necessary variables for x is a hedge formed for $Q[\{s\}]$.

Lemma 11. *Let $\mathcal{G}$ be a semi-Markovian graph such that the edge induced subgraphs of $\mathcal{G}$ over its directed edges and over its bidirected edges are trees. For an arbitrary vertex s and a vertex $x \in Hhull(s) \setminus \{s\}$, $NC_s(x)$ is a hedge formed for $Q[\{s\}]$ in $\mathcal{G}$.*

All of the proofs are provided in Appendix B. One observation is that to solve the min-cost intervention, we can enumerate $NC_s(x)$ for every $x \in Hhull(s)$ and solve the hitting set problem for these hedge. Although the number of such hedges is exactly $|Hhull(s)| - 1$, the hitting set problem is still complex to solve. However, we can further reduce the complexity of the problem as follows. First note that if $y \in NC_s(x)$, by definition of $NC_s(\cdot)$, $NC_s(y) \subseteq NC_s(x)$. Therefore, when considering the hitting set problem, if $NC_s(y)$ is hit, $NC_s(x)$ will also be hit. As a result, we can eliminate $NC_s(x)$ from the sets we are considering. Using the same argument, we begin with some random ordering over the variables $Hhull(S)$ and for every $x \in Hhull(S)$, if x appears in $NC_s(y)$ for some $y \in Hhull(S)$ and the set $NC_s(x)$ is not eliminated yet, we eliminate $NC_s(y)$. At the end of this procedure, we are left with a collection of hedges $\mathbf{F}$ that satisfies the following properties.

1. The min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$ is the min-cost hitting set solution to $\{F \setminus \{s\} | F \in \mathbf{F}\}$.
2. For any two hedges $F, F' \in \mathbf{F}$, $F \cap F' = \{s\}$.
3. $|\mathbf{F}| \leq |Hhull(s)|$[13].

[13] For a formal proof of these properties, refer to the proof of Lemma 12.

Algorithm 5 Polynomial algorithm for bounded hedges.

1: $\mathcal{H} \leftarrow$ empty undirected graph on $V \setminus \mathbf{pa}^{\leftrightarrow}(S)$
2: **for** any pair of vertices $\{a, b\} \subseteq V \setminus (S \cup \mathbf{pa}^{\leftrightarrow}(S))$ **do**
3: **if** $\{a, b\} \cup S$ is a hedge formed for $Q[S]$ **then**
4: draw an edge between a and b in $\mathcal{H}$
5: $A \leftarrow$ the min-weight vertex cover for $\mathcal{H}$
6: **return** $A \cup \mathbf{pa}^{\leftrightarrow}(S)$

Now we observe that the collection $\mathbf{F}$ of hedges are mutually disjoint, and thus the minimum hitting set is simply the union of the minimum cost vertex in each hedge. The following result indicates the correctness and the time complexity of algorithm 4, under the assumption that $\mathcal{G}$ has a tree-like structure.

Lemma 12. *Let $\mathcal{G}$ be a semi-Markovian graph over V such that the edge induced subgraphs of $\mathcal{G}$ over its directed edges and over its bidirected edges are trees. For a vertex s in $\mathcal{G}$, Algorithm 4 returns the min-cost intervention to identify $Q[\{s\}]$ in $\mathcal{G}$ in time $\mathcal{O}(|V|^3)$.*

There are further considerations to Algorithm 4 that we would like to mention. The first one is that this algorithm together with the definitions of nec_s and NC_s, suggest an alternative formulation of the min-cost intervention problem, which is taking into account the set of variables that must be combined together with each variable x to form a hedge for $Q[S]$. The definition of $nec_s(x)$ can be generalized to the case that $\mathcal{G}$ is not a tree anymore, although $nec_s(x)$ will not be a set anymore, but a collection of sets where if an intervention is made upon at least one vertex of all of these sets, no remaining hedge formed for $Q[\{s\}]$ includes x. This indeed suggests a method to enumerate the hedges formed for $Q[S]$ in $\mathcal{G}$. As we saw in this section, for tree-like structures, this enumeration can be executed in polynomial time. However, in general structures, this enumeration method would still take exponential time in the worst case. Another point to mention is that one-step generalizations of the assumption of tree-ness and Algorithm 4 can be thought of, such as the assumption that the number of paths between each pair of vertices in $\mathcal{G}$ is at most 2 (or k, where k is a constant.) Although the tree assumption made in this section might appear restrictive, such generalizations might yield efficient solutions of the min-cost intervention problem that can be used in practice.

C.2. Bounded hedge size

Following the hitting set formulation for the min-cost intervention problem, the two main challenges were enumerating the hedges and solving the hitting set problem afterwards. For a hedge F formed for $Q[S]$ in $\mathcal{G}$, let $(|F| - |S|)$ be the size of this hedge, which is exactly the size of the set to be hit in the hitting set equivalent. If an upper bound on the size of the hedges formed for $Q[S]$ such as $(|F| - |S|) \leq k$ is know where k is a constant, then the task of enumerating the hedges can be performed in polynomial time, as we only need to check the subsets of up to size k. Note that as discussed in Section 3.2, this argument is still valid if the upper bound works for the set of *minimal* hedges. However, the hitting set task still remains exponential in the worst case. On the other hand, for certain values of k, the min-cost intervention problem can be solved in polynomial time without using the hitting set formulation. For $k = 1$, the set of minimal hedges formed for $Q[S]$ reduces to the hedge structures composed of S and one variable in $\mathbf{pa}^{\leftrightarrow}(S)$. From lemma 4, we know that in such a structure, the optimal intervention is $A^* = \mathbf{pa}^{\leftrightarrow}(S)$, and $\mathbf{pa}^{\leftrightarrow}(S)$ can be constructed in linear time. In this section, we show that for $k = 2$, that is, given that every minimal hedge formed for $Q[S]$ has size at most 2, the min-cost intervention problem can be solved in polynomial time. We begin with the following property of the formed hedges, which will help us model the min-cost intervention problem as a maximum matching problem in a bipartite graph through Konig's theorem (Konig, 1931).

Lemma 13. *Let $\mathcal{G}$ be a semi-Markovian graph and S be a subset of its vertices such that $\mathcal{G}_{[S]}$ is a c-component. Construct an undirected graph $\mathcal{H}$ on the same set of vertices as $\mathcal{G} \setminus \mathbf{pa}^{\leftrightarrow}(S)$ as follows. For any hedge of size 2 formed for $Q[S]$ such as F, connect the two vertices in $F \setminus S$ with an edge. The resulting graph $\mathcal{H}$ is bipartite.*

First, note that for any hedge F formed for $Q[S]$ in $\mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]}$, there exists an edge between the two vertices $F \setminus S$ in the undirected graph $\mathcal{H}$. Since the min-cost intervention to identify $Q[S]$ in $\mathcal{G}$ is the union of $\mathbf{pa}^{\leftrightarrow}(S)$ and the minimum hitting set for the sets $F \setminus S$ (Lemma 3), the min-cost intervention can also be given as the union of $\mathbf{pa}^{\leftrightarrow}(S)$ and the minimum vertex cover for the undirected graph $\mathcal{H}$. Lemma 13 states that $\mathcal{H}$ is bipartite. It is known that in bipartite graphs, the minimum-weight vertex cover problem is equivalent to a maximum matching (when the costs are uniform), or a maximum

Algorithm 6 Polynomial time algorithm for special $\mathbf{C}(\cdot)$.

```
initialize I ← ∅
if I ∈ ID_1(S) then
    return I
V' ← ancestors of S in 𝒢, {v_1, ..., v_k}
sort the vertices V' s.t. C(v_i) < C(v_{i+1}) ∀1 ≤ i < k
while true do
    i ← 0
    I' ← I
    while true do
        i ← i + 1
        I' ← I' ∪ {v_i}
        if I' ∈ ID_1(S) then
            break
    I ← I ∪ {v_i}
    if I ∈ ID_1(S) then
        return I
```

flow problem (when the costs are not uniform) (Konig, 1931). There are various polynomial time algorithms to solve these problems, such as Ford-Fulkerson, Edmonds-Karp and push-relabel algorithms to name a few (Ford & Fulkerson, 1956; Edmonds & Karp, 1972; Goldberg & Tarjan, 1988). Consequently, under the assumption that for any minimal hedge F formed for $Q[S]$, $|F| - |S| \leq 2$, we propose Algorithm 5 to solve the min-cost intervention problem in polynomial time. Any appropriate algorithm can be used as a subroutine in line (5) of Algorithm 5.

C.3. Special cost functions

We have discussed special graph structures so far. However, in certain cases, knowledge about the form of the cost function $\mathbf{C}(\cdot)$ can help us solve the min-cost intervention problem efficiently. One such case is when the costs of intervening on variables are far enough from each other. As a concrete example, let the vertices of $\mathcal{G}$ be $v_1, ..., v_n$, with the cost function $\mathbf{C}(v_i) = 2^i$ for $1 \leq i \leq n$. We begin with testing the sets $\{v_1\}, \{v_1, v_2\}, ..., \{v_1, v_2, ..., v_n\}$, until we reach at the first set $I_j = \{v_1, ..., v_j\} \in \mathbf{ID_1}(S)$. Since the cost of this intervention is $\mathbf{C}(I_j) = \sum_{i=1}^{j} 2^i < 2^{j+1}$, the min-cost intervention does not include any of the variables $v_{j+1}, ..., v_n$, as the cost of any of these variables is at least 2^{j+1}. Further, as intervening on more variables cannot induce new hedges, and by definition of I_j, no subset of $\{v_1, ..., v_{j-1}\}$ is in $\mathbf{ID_1}(S)$. This implies that if A^* is a min-cost intervention to identify $Q[S]$ in $\mathcal{G}$, then $v_j \in A^*$ and $v_l \notin A^*$ for any $l > j$. We then restart the procedure, testing the sets $\{v_1\} \cup \{v_j\}, \{v_1, v_2\} \cup \{v_j\}, ..., \{v_1, ..., v_{j-1}\} \cup \{v_j\}$ to find the first set $I_k = \{v_1, ..., v_k\} \cup \{v_j\} \in \mathbf{ID_1}(S)$. Again with the same arguments, we can conclude that $v_k \in A^*$ and $v_l \notin A^* \setminus \{v_j\}$ for any $l > k$. Continuing in the same manner, we construct the min-cost vertex cover (which is unique in this setting) after $\mathcal{O}(|V|)$ iterations in the worst case. Each iteration tests whether a set is a hedge at most $|V|$ times, which can be performed using two depth-first searches ($\mathcal{O}(|V|^2)$). As a result, the min-cost intervention can be solved in time $\mathcal{O}(|V|^4)$ in the worst case.

Note that the property we used throughout our reasoning was the fact that having sorted the variables based on their intervention costs as $v_1, ..., v_n$, for any $1 \leq j < n$, $\mathbf{C}(\{v_1, ..., v_i\}) < \mathbf{C}(v_{i+1})$. With such a cost function, Algorithm 6 solves the min-cost intervention problem in time $\mathcal{O}(|V|^4)$ in the worst case, i.e., regardless of the structure of $\mathcal{G}$. Note that this algorithm has the same worst-case time complexity for both when $\mathcal{G}_{[S]}$ is a c-component and when it is not. Also note that as an optional step, we can begin with constructing the hedge hull of S in $\mathcal{G}_{V \setminus \mathbf{pa}^{\leftrightarrow}(S)}$ (denoted by H) if $\mathcal{G}_{[S]}$ is a c-component. In this case, sorting the variables in H based on their cost as $h_1, ..., h_m$, we only need the assumption that $\mathbf{C}(\{h_1, ..., h_i\}) < \mathbf{C}(h_{i+1})$ for $1 \leq i \leq m-1$, and the worst-case time complexity would be $\mathcal{O}(|V|^3 + |H|^4)$.

Lemma 14. *Let $\mathcal{G}$ be a semi-Markovian graph on vertices V, along with a cost function $\mathbf{C}(\cdot)$. Let S be subset of vertices of $\mathcal{G}$, and $V' = \{v_1, ..., v_k\}$ be the set of ancestors of S in $\mathcal{G}$. If $\mathbf{C}(\{v_1, ..., v_i\}) < \mathbf{C}(v_{i+1})$ for every $1 \leq i < k$, then Algorithm 6 returns the min-cost intervention to identify $Q[S]$ in $\mathcal{G}$ in time $\mathcal{O}(|V|^4)$.*

Algorithm 7 Heuristic algorithm 1.

1: **input:** $\mathcal{G}, S, \mathbf{C}(\cdot)$, **output:** $A \in \mathbf{ID_1}(S)$
2: $H \leftarrow Hhull(S, \mathcal{G}_{V\setminus \mathbf{pa}^{\leftrightarrow}(S)})$
3: Build $\mathcal{H}$ on $H \cup \{x, y\}$: draw an undirected edge between $v_1, v_2 \in H \setminus S$ if there is a bidirected edge between them in $\mathcal{G}$. Connect x to $\mathbf{pa}(S) \cap H$ and y to S.
4: $MC \leftarrow$ minimum weight vertex cut for $x - y$ in $\mathcal{H}$, with weights $\omega(v) = \mathbf{C}(v)$ for $v \notin S$ & $\omega(s) = \infty$ for $s \in S$
5: $A \leftarrow MC \cup \mathbf{pa}^{\leftrightarrow}(S)$
6: **return** A

Algorithm 8 Heuristic algorithm 2.

1: **input:** $\mathcal{G}, S, \mathbf{C}(\cdot)$, **output:** $A \in \mathbf{ID_1}(S)$
2: $H \leftarrow Hhull(S, \mathcal{G}_{V\setminus \mathbf{pa}^{\leftrightarrow}(S)})$
3: Build $\mathcal{H}$ on $H \cup \{x, y\}$: for $v_1, v_2 \in H \setminus S$, draw $v_1 \rightarrow v_2$ in $\mathcal{H}$ if this edge exists in $\mathcal{G}$. Draw the edges from x to $\mathbf{pa}(S) \cap H$ and from S to y
4: $MC \leftarrow$ minimum weight vertex cut for $x - y$ in $\mathcal{H}$, with weights $\omega(v) = \mathbf{C}(v)$ for $v \notin S$ & $\omega(s) = \infty$ for $s \in S$
5: $A \leftarrow MC \cup \mathbf{pa}^{\leftrightarrow}(S)$
6: **return** A

D. Heuristic Algorithms

In this section, we first present the three heuristic algorithms proposed in Section 3.5. We discuss their correctness, their running times, and how they compare to each other. Later, we propose a polynomial-time improvement that can be utilized as a post-process to improve the output of these algorithms.

The first heuristic algorithm is depicted as Algorithm 7. We begin with removing $\mathbf{pa}^{\leftrightarrow}(S)$ from the graph, as we already know that this set must be included in the output. We then build an undirected graph $\mathcal{H}$ over the vertices of $H = Hhull(S, \mathcal{G}_{[V\setminus \mathbf{pa}^{\leftrightarrow}(S)]})$, along with two extra vertices x and y. For every bidirected edge $\{v_1, v_2\}$ in $\mathcal{G}_{[H]}$, we draw a corresponding edge between v_1 and v_2 in $\mathcal{H}$. Finally, we connect x to $\mathbf{pa}(S) \cap H$ and y to S with an edge. Note that every undirected path between x and y in $\mathcal{H}$ corresponds to a bidirected path that connects a vertex in S to a vertex in $\mathbf{pa}(S) \cap H$ in $\mathcal{G}$. If we intervene on a subset of variables A such that no such path exists anymore, the hedge hull of S in the remaining graph will be S itself, as none of the vertices $\mathbf{pa}(S)$ are in the same c-component of S. Consequently, the effect $Q[S]$ becomes identifiable. With that being said, we solve for the minimum-weight vertex cut for $x - y$ in $\mathcal{H}$ in line (4) of the algorithm. We set the weights of the vertices in S to infinity to ensure that we do not intervene on them. Note that the min-weight vertex cut in an undirected graph can be turned into an equivalent problem in a directed graph, by simply substituting every undirected edge with two directed edges in the opposite direction. Further, min-weight vertex cut can be reduced to min-weight edge cut through a trivial reduction: We replace every vertex v with two vertices v_1, v_2, add an edge from v_1 to v_2 with the same weight as the weight of v in the original graph, and connect every edge that goes into v to v_1, and every edge that goes out of v to v_2. The resulting problem can be solved using any of the standard max-flow-min-cut algorithms. We used the push-relabel algorithm to solve the max-flows throughout our simulations (Goldberg & Tarjan, 1988).

The second heuristic algorithm, depicted as Algorithm 8, relies on similar ideas. Again, we begin with removing $\mathbf{pa}^{\leftrightarrow}(S)$ from the graph, as we already know that this set must be included in the output. We then build a directed graph $\mathcal{H}$ over the vertices of $H = Hhull(S, \mathcal{G}_{[V\setminus \mathbf{pa}^{\leftrightarrow}(S)]})$, along with two extra vertices x and y. For every directed edge $v_1 \rightarrow v_2$ in $\mathcal{G}_{[H]}$, we draw a corresponding edge between $v_1 \rightarrow v_2$ in $\mathcal{H}$. Finally, we draw an edge from x to all vertices in $\mathbf{biD}(S) \cap H$ and from all vertices in S to y. Note that every directed path from x to y in $\mathcal{H}$ corresponds to a directed path that connects a vertex in $\mathbf{biD}(S)$ to a vertex in S in $\mathcal{G}$. If we intervene on a subset of variables A such that no such path exists anymore, the hedge hull of S in the remaining graph will be S itself, as none of the vertices $\mathbf{biD}(S)$ have a directed path to S. Consequently, the effect $Q[S]$ becomes identifiable. With that being said, we solve for the minimum-weight vertex cut for $x - y$ in $\mathcal{H}$ in line (4) of the algorithm. We set the weights of the vertices in S to infinity to ensure that we do not intervene on them. As mentioned above, we reduce the min-weight vertex cut to min-weight edge cut, and then use max-flow algorithms to solve it.

Finally, we proceed to our third heuristic algorithm, which is based on a greedy approach. First, note that if we intervene on every variable in the hedge hull of S except S, $Q[S]$ becomes identifiable. That is, defining $H = Hhull(S, \mathcal{G}_{[V\setminus \mathbf{pa}^{\leftrightarrow}(S)]})$, one trivial set in $\mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$ is $\{(H \setminus S) \cup \mathbf{pa}^{\leftrightarrow}(S)\}$. Similarly, if we intervene on a set of variables A, then $A \cup Hhull(S, \mathcal{G}_{[V \setminus A]} \setminus S)$ is a trivial solution. In our greedy approach, we minimize the cost of this trivial solution at each iteration. We proceed as follows. We maintain an intervention set A, which is initialized as $\mathbf{pa}^{\leftrightarrow}(S)$. At each iteration, we find the vertex $x \in Hhull(S, G_{[V \setminus A]}) \setminus S$ that minimizes the objective function

$$f(x) = \mathbf{C}(x) + \mathbf{C}(Hhull(S, \mathcal{G}_{[V \setminus (A \cup \{x\})]})),$$

and add this vertex to A. Note that the function $f(x)$ is exactly the cost of the trivial solution in graph $\mathcal{G} \setminus (A \cup \{x\})$. We add one vertex in each iteration until we reach a point where $Q[S]$ becomes identifiable. The following result indicates the correctness of Algorithm 9 along with its computational complexity.

Lemma 15. *Given a semi-Markovian graph $\mathcal{G}$ on V and a subset of its vertices S such that $\mathcal{G}_{[S]}$ is a c-component, Algorithm 9 returns a set A such that $\{A\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$ in time $\mathcal{O}(|V|^5)$ in the worst case.*

Algorithm 9 Heuristic greedy algorithm.

$H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]})$
initialize $A \leftarrow \mathbf{pa}^{\leftrightarrow}(S)$
while $H \neq S$ **do**
    $c_{min} \leftarrow \mathbf{C}(H)$
    $i \leftarrow null$
    for $v \in H$ **do**
        $H' \leftarrow Hhull(S, \mathcal{G}_{[H \setminus \{v\}]})$
        if $\mathbf{C}(H') + \mathbf{C}(v) \leq c_{min}$ **then**
            $c_{min} \leftarrow \mathbf{C}(H') + \mathbf{C}(v)$
            $i \leftarrow v$
    $H \leftarrow Hhull(S, \mathcal{G}_{[H \setminus \{i\}]})$
    $A \leftarrow A \cup \{i\}$
return A

General subset identification using heuristic algorithms. The heuristic algorithms proposed in this work are devised under the assumption that $\mathcal{G}_{[S]}$ is a c-component. However, as claimed in the main text, all of the three heuristic algorithms return a valid intervention set to identify $Q[S]$ in $\mathcal{G}$, even if $\mathcal{G}_{[S]}$ is not a c-component. This follows from the result that $Q[S]$ is identifiable in $\mathcal{G}$, if and only if $Q[S_1], ..., Q[S_k]$ are identifiable in $\mathcal{G}$, where $S_1, ..., S_k$ are the maximal c-components of $\mathcal{G}_{[S]}$ (Tian & Pearl, 2002). The following result formalizes this claim.

Lemma 16. *Given a semi-Markovian graph on V and a subset S of its vertices, Algorithms 7, 8 and 9 return a subset A of the vertices of $\mathcal{G}$ such that $\{A\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$, in time $\mathcal{O}(|V|^3)$, $\mathcal{O}(|V|^3)$ and $\mathcal{O}(|V|^5)$, respectively.*

Note that we Lemma 16 does not require that $\mathcal{G}_{[S]}$ be a c-component, unlike Lemmas 8 and 9. As a result, all of these algorithms can also be utilized as a subroutine in line (6) of Algorithm 3, the general algorithm proposed in this work.

Post-process. In many cases, when the output of the proposed heuristic algorithms is not optimal, it is a super-set of the optimal intervention. As a result, we propose greedily deleting such extra variables from the intervention set A while $Q[S]$ remains identifiable. That is, assuming A is the output of one of the Algorithms 7,8,9, we start with the vertex $a \in A$ with the highest cost, and while there exists $a \in A \setminus \mathbf{pa}^{\leftrightarrow}(S)$ such that $\{A \setminus \{a\}\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$, we remove a from A. Testing whether a set is in $\mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$ requires time $\mathcal{O}(|V|^3)$ in the worst case. As a result, the proposed post-process does not alter the worst-case complexity of the algorithms.

Discussion. The proposed algorithms have no theoretical guarantee of how well they can approximate the solution to the min-cost intervention problem. However, their performances as well as their runtimes are dependent on the structure of the graph $\mathcal{G}_{[H]}$, where $H = Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]})$. For instance, if the edge-induced subgraph of $\mathcal{G}_{[H]}$ on its bidirected edges is much more dense than the edge-induced subgraph of $\mathcal{G}_{[H]}$ on its directed edges, Algorithm 7 will need to solve a more complex min-weight vertex cover problem compared to Algorithm 8. It will also add potentially many extra vertices that are not needed in the intervention set. Since $\mathcal{G}_{[H]}$ is constructed as a pre-process of all three algorithms, we propose choosing the heuristic algorithm after constructing $\mathcal{G}_{[H]}$ as follows. Algorithm 8 is preferred over the other two, as it solves a min vertex cut in a directed graph rather than an undirected graph. However, if the graph $\mathcal{G}_{[H]}$ is dense on its directed edges, we choose Algorithm 7. In certain cases, as shown by our empirical evaluation, the greedy approach achieves lower regret despite the higher time complexity.

E. Hitting Set & Algorithm 2

E.1. Greedy approach for minimum hitting set

In this section, we present the greedy weighted minimum hitting set algorithm mentioned in the main text (Johnson, 1974). This greedy approach is depicted in Algorithm 10. Let V, $\mathbf{F}$, and $\omega(\cdot)$ be the universe of objects, the collection of sets for which we want to find a hitting set, and the weight function respectively. For an object $v \in V$, we denote by $N(v)$ the number of sets $F \in \mathbf{F}$ such that $v \in F$, that is, the number of sets v hits. We begin with an empty hitting set A. At each iteration, we choose the variable $v \in V$ that maximizes $\frac{N(v)}{\omega(v)}$, and add it to A. We then remove all the sets F that include v from $\mathbf{F}$. The algorithm runs until $\mathbf{F}$ becomes empty. The resulting set A is a hitting set for $\mathbf{F}$. It has been shown that this greedy algorithm achieves a logarithmic-factor approximation of the optimal hitting set in the worst case (Johnson, 1974; Chvatal, 1979). Note that using certain data structures, we can avoid recalculating $N(v)$ at each iteration in line (4).

Algorithm 10 Greedy weighted minimum hitting set algorithm.

1: **input:** universe V, collection of sets $\mathbf{F}$, weights $\omega(v)$ for $v \in V$, **output:** a hitting set for $\mathbf{F}$
2: **while** $\mathbf{F} \neq \emptyset$ **do**
3:     **for all** $v \in V$ **do**
4:         $N(v) \leftarrow |\{F \in \mathbf{F} | v \in F\}|$
5:     $v \leftarrow \arg\min_{v \in V} \frac{N(v)}{\omega(v)}$
6:     $A \leftarrow A \cup \{v\}$
7:     $\mathbf{F} \leftarrow \mathbf{F} \setminus \{F \in \mathbf{F} | v \in F\}$
8: **return** A

E.2. On Algorithm 2

In this section, we provide a slight modification of Algorithm 2. One caveat to Algorithm 2 is that it might call numerous times as a subroutine, a solution to the minimum hitting set problem (line (13)). Although we propose using the greedy approach mentioned above as the subroutine, we also provide a modification, depicted as Algorithm 11, which reduces the number of calls to this subroutine as follows. At the end of each iteration (inner loop, that is, lines (7-13)), instead of solving the minimum hitting set problem, we simply add the vertex a found in the last step to a set of interventions A. We postpone the call to minimum hitting set to when A grows large enough so that $\{A\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$. Through this modification, we discover more hedges and add them to $\mathbf{F}$ before calling for the solution of the minimum hitting set problem. Therefore, this modification reduces the number of calls to the subroutine of solving the min hitting set in certain cases.

F. Further Empirical Evaluation

In this section, we provide further details of the experimental setup of the paper. We also provide complementary evaluations of our proposed algorithms.

Setup. We have evaluated our algorithms in two different settings. In Appendix F.1, we evaluate our algorithms on a set of well-known graphs, which are the benchmark causal graphs in the causality literature. These graphs are obtained under the assumption of no latent variables. However, often the observed variables of a system are confounded by a hidden variable. We added a common confounder for each pair of variables in these graphs with probability q. We then ran our algorithms to find the min-cost intervention for identifying $Q[S]$, where S is the last vertex in the causal order. We assumed that the cost of intervening on each variable is uniformly sampled from $\{1, 2, 3, 4\}$.

In the second setting considered throughout our evaluations, we generated random graphs based on Erdos-Renyi generative model. The directed and bidirected edges of the graph in this model are sampled mutually independently, with probabilities p and q respectively. We then assigned a random cost of intervening to each variable, sampled from the uniform distribution over $\{1, 2, 3, 4\}$. Set S in these set of evaluations is randomly chosen among the last 5% vertices of the graph, such that $\mathcal{G}_{[S]}$

Algorithm 11 Modified algorithm to reduce the calls to minimum hitting set.

$\mathbf{F} \leftarrow \emptyset, A \leftarrow \emptyset$
$H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus \mathbf{pa}^{\leftrightarrow}(S)]})$
if $H = S$ **then**
  return $\mathbf{pa}^{\leftrightarrow}(S)$
while True **do**
  while True **do**
    while True **do**
      $a \leftarrow \arg\min_{a \in H \setminus S} \mathbf{C}(a)$
      if $Hhull(S, \mathcal{G}_{[H \setminus \{a\}]}) = S$ **then**
        $\mathbf{F} \leftarrow \mathbf{F} \cup \{H\}$
        break
      else
        $H \leftarrow Hhull(S, \mathcal{G}_{[H \setminus \{a\}]})$
    $A \leftarrow A \cup \{a\}$
    if $\{A \cup \mathbf{pa}^{\leftrightarrow}(S)\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$ **then**
      break
    $H \leftarrow Hhull(S, \mathcal{G}_{[V \setminus (A \cup \mathbf{pa}^{\leftrightarrow}(S))]})$
  $A \leftarrow$ min hitting set for $\{F \setminus S | F \in \mathbf{F}\}$
  if $\{A \cup \mathbf{pa}^{\leftrightarrow}(S)\} \in \mathbf{ID}_{\mathcal{G}}(S, V \setminus S)$ **then**
    return $A \cup \mathbf{pa}^{\leftrightarrow}(S)$

is a c-component. Appendix F.2 provides empirical results of our algorithms on the randomly generated graphs. Finally, an evaluation of the hedge enumeration task of Algorithm 2 is given in Figure 10.

F.1. Benchmark Structures

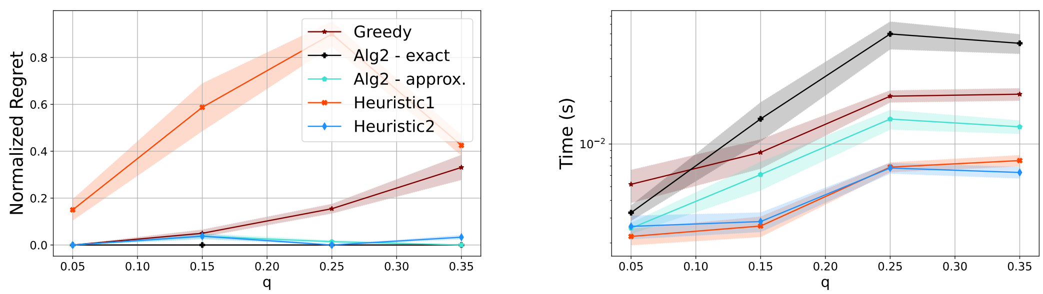
In this section, we evaluate our algorithms on graphs corresponding to real-world problems, namely the Barley (Kristensen & Rasmussen, 1997), Water (Jensen et al., 1989) and Mehra (Vitolo et al., 2018) structures [14]. These structures are formed as causal DAGs under the assumption of no hidden confounder. However, often hidden variables confound observed variables. In our experiments, we randomly added a latent confounder for every pair of variables with probability $q \in \{0.05, 0.15, 0.25, 0.35\}$, and evaluated the performance of our algorithms. The intervention costs are assigned uniformly at random from $\{1, 2, 3, 4\}$, and the set S is chosen to be the last vertex in the causal ordering. The results are depicted in Figure 6.

F.2. Randomly Generated Graphs

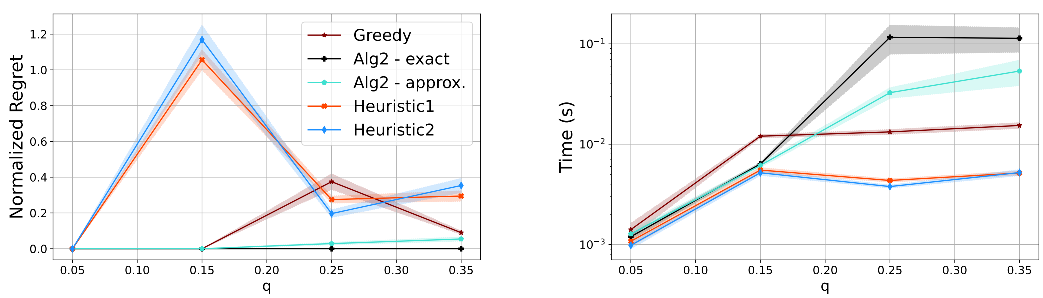
Figure 7 illustrates the runtime and the normalized regret (as defined in Section 5) of our algorithms on randomly generated graphs, with different values of p and q over random graphs of size $n = 10$ to $n = 200$. Figure 8 shows the effect of the density of the bidirected edges on the performance of the algorithms. Random graphs of size $n = 30$ are generated with different values of p. Figure 9 shows the effect of the density of the directed edges on the performance of the algorithms. Random graphs of size $n = 30$ are generated with different values of the parameter q. An important observation in all of these figures is that the normalized regret is not necessarily a monotone function of the graph size. This measure depends on the structure of the graph, size and location of the desired set S, and the random cost assignments. Another observation is that the runtime of the algorithms is not a monotone funciton of the graph density. This is due to the fact that the denser the graph becomes, the larger the set $\mathbf{pa}^{\leftrightarrow}(S)$ grows. As a result, the set H defined in Equation 3 becomes smaller and after a certain threshold, the problem becomes even simpler for denser graphs.

Figure 10 demonstrates how Algorithm 2 circumvents the hedge enumeration task by enumerating only a small portion of them. We have plotted the number of hedges formed for $Q[S]$ in random graphs of different sizes generated with parameters $p = 0.35$ and $q = 0.25$, as opposed to the number of hedges that Algorithm 2 discovers before finding the optimal min-cost intervention solution. The number of hedges formed for $Q[S]$ is counted after removing $\mathbf{pa}^{\leftrightarrow}(S)$.

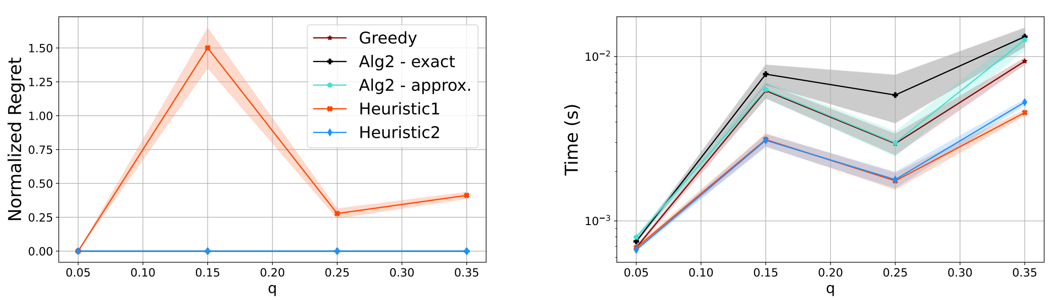
[14] See https://www.bnlearn.com/bnrepository/ for details.



(a) Barley structure

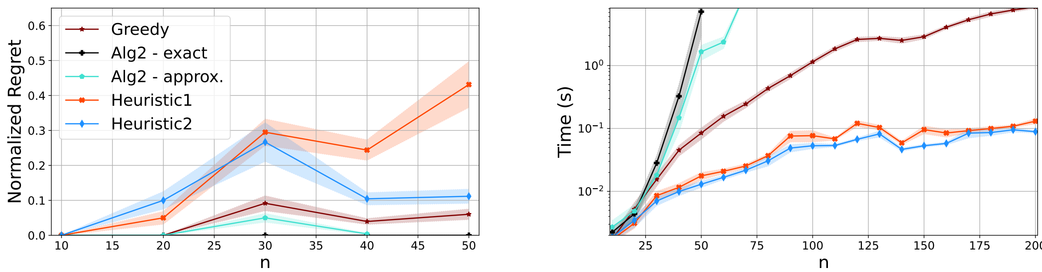


(b) Water structure

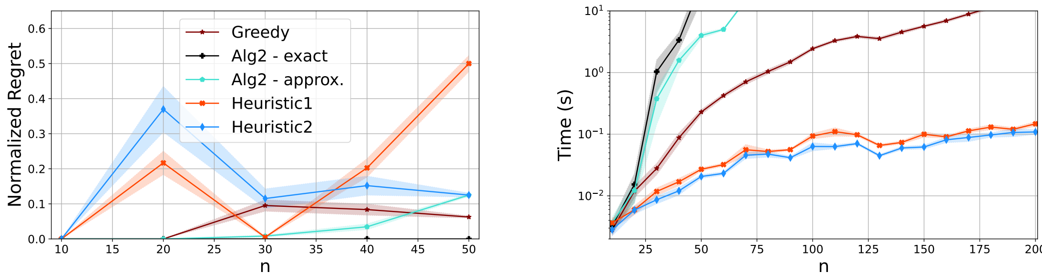


(c) Mehra structure

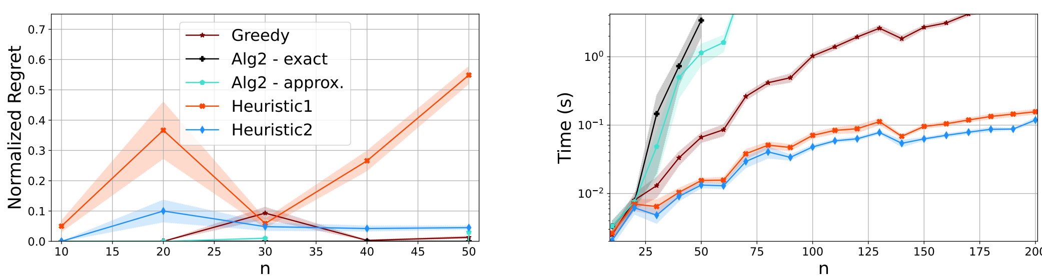
Figure 6. The performance of the proposed algorithms on three real-world structures.



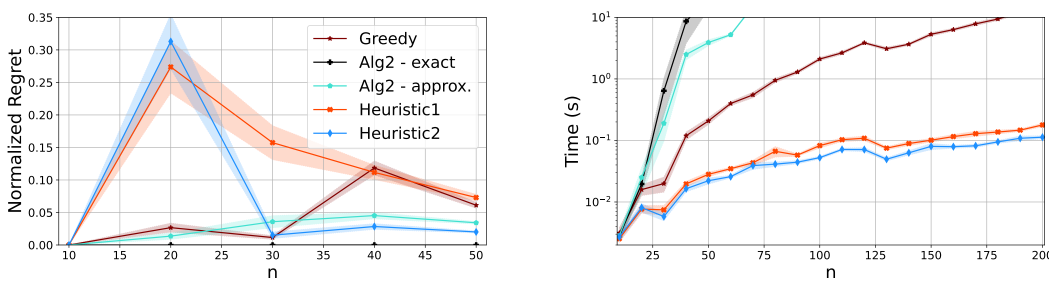
(a) $p = 0.15, q = 0.15$



(b) $p = 0.25, q = 0.15$

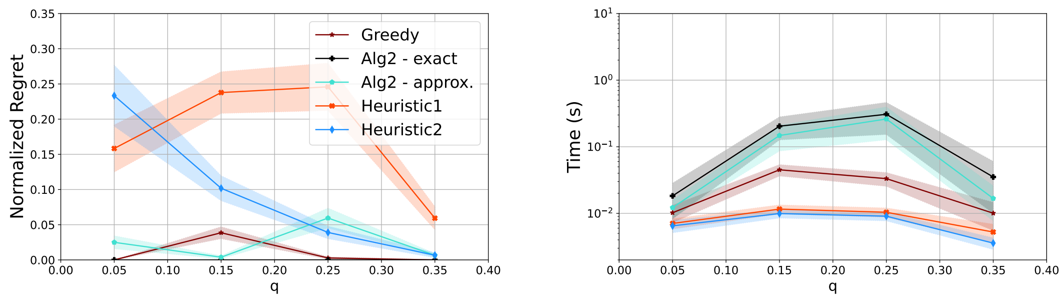


(c) $p = 0.15, q = 0.25$

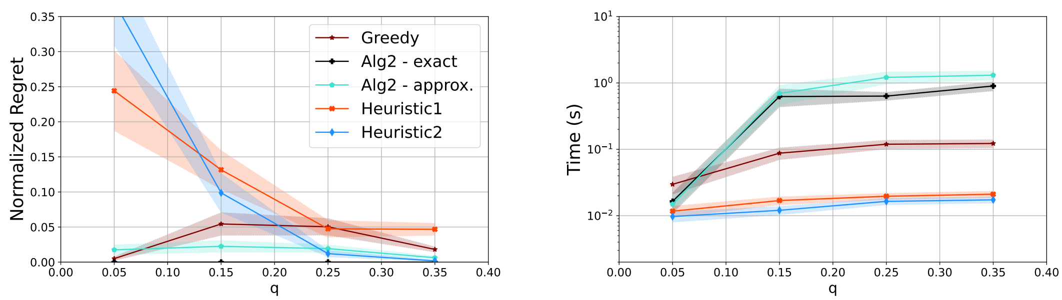


(d) $p = 0.25, q = 0.25$

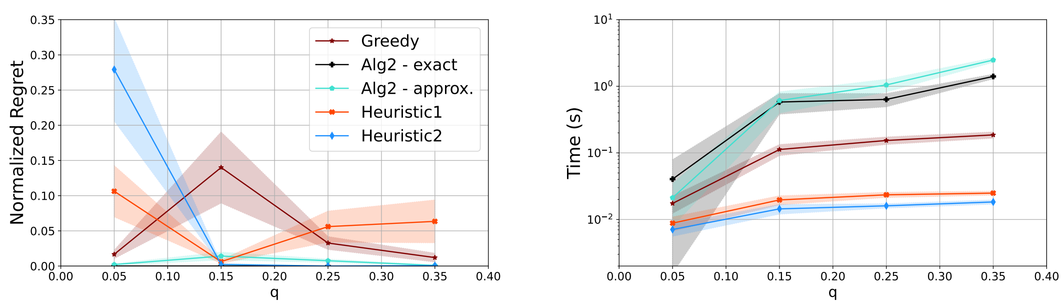
Figure 7. Evaluation of the proposed algorithms on random graphs with various parameters.



(a) $p = 0.15$

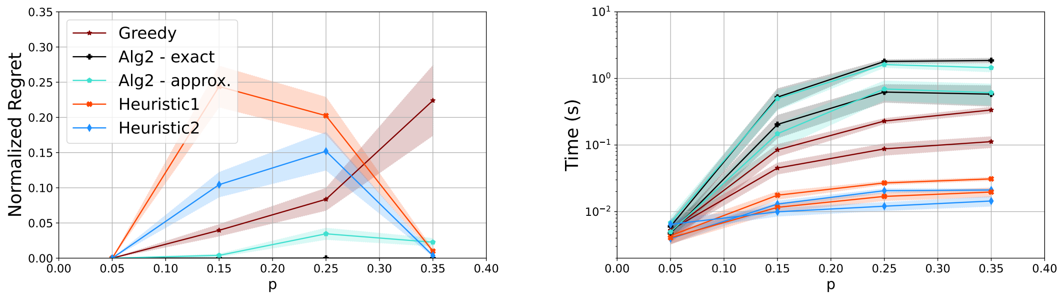


(b) $p = 0.25$

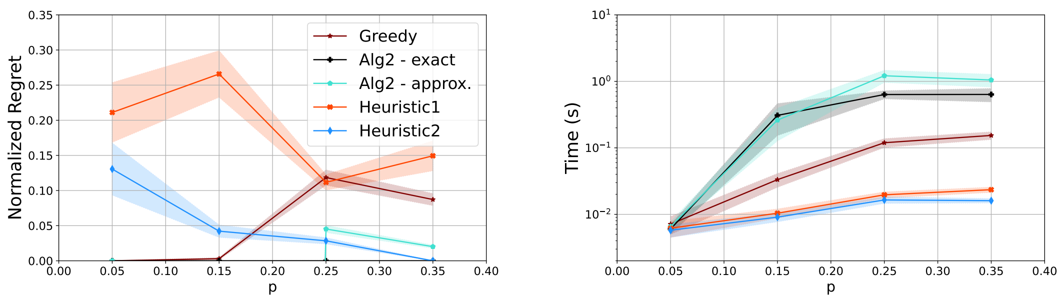


(c) $p = 0.35$

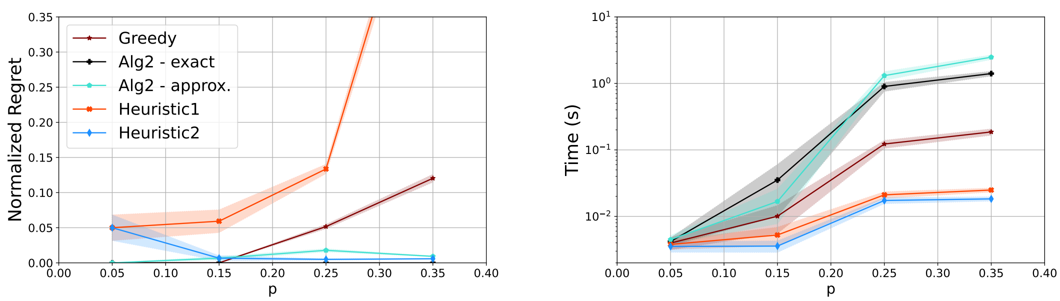
Figure 8. The effect of the density of bidirected edges. Random graphs of size $n = 30$ are generated with different densities of directed edges.



(a) $q = 0.15$



(b) $q = 0.25$



(c) $q = 0.35$

Figure 9. The effect of the density of directed edges. Random graphs of size $n = 30$ are generated with different densities of bidirected edges.

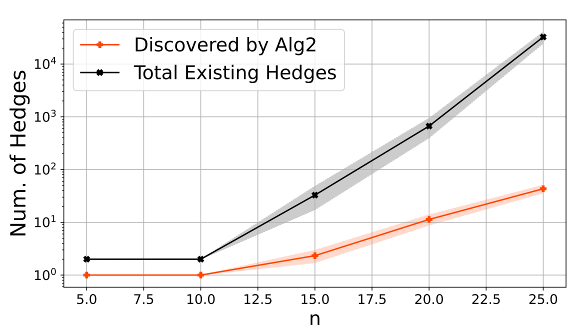


Figure 10. Number of the total hedges formed for $Q[S]$ vs the number of hedges Algorithm 2 discovers until finding the optimal solution.