Causal Discovery in Probabilistic Networks with an Identifiable Causal Effect

Anonymous Author(s)
Affiliation
Address
email

Abstract

Causal identification is at the core of the causal inference literature, where complete algorithms have been proposed to identify causal queries of interest. The validity of these algorithms hinges on the restrictive assumption of having access to a correctly specified causal structure. In this work, we study the setting where a probabilistic model of the causal structure is available. Specifically, the edges in a causal graph are assigned probabilities which may, for example, represent degree of belief from domain experts. Alternatively, the uncertainly about an edge may reflect the confidence of a particular statistical test. The question that naturally arises in this setting is: Given such a probabilistic graph and a specific causal effect of interest, what is the subgraph which has the highest plausibility and for which the causal effect is identifiable? We show that answering this question reduces to solving an NP-hard combinatorial optimization problem which we call the edge ID problem. We propose efficient algorithms to approximate this problem, and evaluate our proposed algorithms against real-world networks and randomly generated graphs.

1 Introduction

A large proportion of questions of interest in various fields including but not limited to psychology, social sciences, behavioural sciences, medical research, epidemiology, economy, etc. are causal in nature [21, 13, 2]. In order to estimate causal effects, the gold standard is performing controlled interventions and experiments. Unfortunately, such experiments can be prohibitively expensive, unethical, or impractical (consider, for example, an experiment in which participants are required to smoke in order to understand the links to cancer) [3, 5]. In contrast, non-experimental data are comparatively abundant, and no expensive interventions are required to generate such data. This has motivated the development of numerous techniques for understanding whether a causal query can be answered using observational data. Specifically, if a particular causal query is *identifiable*, it means it can be expressed as a function of the observational distribution, and thus estimated from observational data (at least in principle).

A significant body of the causal inference literature is dedicated to the identification problem [18, 13, 16, 7, 12]. In particular, Huang and Valtora presented a complete algorithmic approach to decide the identifiability of a specific query, and proved that Pearl's do calculus is complete, in the sense that if a causal query is identifiable, a sequence of do calculus rules can be applied to derive an identification expression for that query [6]. Furthermore, Shpitser and Pearl provided a graphical criteria to decide the identifiability, based on the *hedge* criterion [16]. However, all of these results hinge on full specification of the causal structure, i.e., access to a correctly specified Acyclic Directed Mixed Graph (ADMG) that models the causal dynamics of the system. This requirement is restrictive in a number of ways. Firstly, the causal identification problem is concerned with inference from the observational data, but the ADMG cannot be inferred from the observational distribution alone.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Secondly, structure learning methods rely heavily on statistical tests, which are prone to errors arising from lack of sufficient data and method-specific limitations [15] which can result in misspecification of the causal structure.

As opposed to full specification of the causal structure, we propose the setting in which we only have access to a probabilistic model of the causal structure. For instance, an ADMG $\mathcal{G}$ is given along with probabilities assigned to each edge of $\mathcal{G}$. An example is shown in Figure 1a. These probabilities could represent uncertainties arising from statistical tests, or the strength of belief of domain experts concerning the plausibility of the existence of an edge. Under this setting, each ADMG on the set of vertices of $\mathcal{G}$ is assigned its own plausibility score. Since the causal structure is not deterministic anymore, answering questions such as "*is the causal effect $P_X(Y)$ identifiable?*" also becomes probabilistic in nature. One can compare the overall plausibility of different subgraphs in which the causal effect is identifiable, and then select the graph which maximises the plausibility. Indeed, identification is often assumed on the basis of ignorability (i.e., no unobserved confounders exist) [8, 14], thus the use of probabilistic models enables us to quantify the strength of such an assumption, as well as to relax it by selecting the most plausible subgraph in which the assumption holds true.

In this work, for a specific causal query $P_X(Y)$, we first answer the question "which graph has the highest plausibility among those compliant with the probabilistic ADMG model that renders $P_X(Y)$ identifiable?" The answer to this question then shows us with what confidence we can carry out the causal identification task using the combination of the data at hand and the corresponding probabilistic model.

Noting that the causal identification task is carried out through an identification formula which is based on the causal structure, our second focus is on deriving an identification formula for a given causal query that holds with the highest probability. This problem is different from the former in the sense that a single identification formula can be valid with respect to a set of different graphs. Therefore, the probability that a given identification formula is valid for a causal query would be the aggregate probability of all graphs on which this formula is valid. We shall illustrate this point in more detail through Example 1 in Section 2. To identify the most probable identification formula, we first show that if an identification formula is valid w.r.t. a causal graph, it is also valid w.r.t. all its edge-induced subgraphs. Afterwards, we propose a surrogate problem (see Problem 2 in Section 2.1) that recovers a causal graph with highest aggregated probability of its subgraphs. Both problems discussed in this work are aimed at evaluating the plausibility of performing causal identification for a specific query given a dataset and a non-deterministic model describing the causal structure.

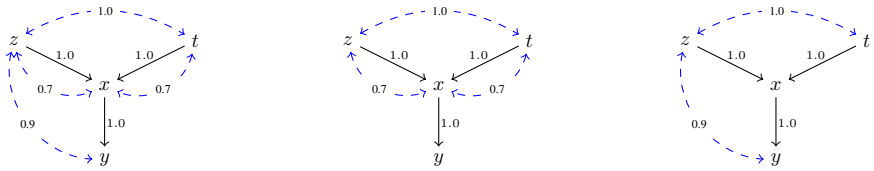
To sum up, our main contributions are as follows.

1. We study the problem of causal identifiability in probabilistic causal models, where there are uncertainties about the existence of edges and whether a given causal effect is identifiable. More precisely, we consider two problems: 1) finding the most probable graph that renders a desired causal query identifiable, and 2) finding the graph with the highest aggregate probability over its edge-induced subgraphs that renders a desired causal query identifiable.
2. We show that both aforementioned problems reduce to a special combinatorial optimization problem which we call the *edge ID problem*. We prove that the edge ID problem is NP-hard, and thus, so are both of the problems we discussed.
3. We propose several exact and heuristic algorithms for the aforementioned problems.

In Section 2, we introduce the terminology and formally define the two problems we are considering in this work. In Section 3, we show that both of these problems are equivalent to the edge ID problem. Furthermore, we show that the edge ID problem is NP-hard. We discuss algorithmic approaches (both exact and heuristic) in Section 4. Empirical evaluations of our algorithms are presented in Section 5. Proofs and accompanying code are provided in the appendices and in supplementary material, respectively.

2 Preliminaries

We utilise small letters for variables, and capital letters for sets of variables. Calligraphic letters are used to denote graphs. An acyclic directed mixed graph (ADMG) $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$ is defined as an acyclic graph on the vertices $V^{\mathcal{G}}$, where $E_d^{\mathcal{G}} \subseteq V^{\mathcal{G}} \times V^{\mathcal{G}}$ and $E_b^{\mathcal{G}} \subseteq \binom{V^{\mathcal{G}}}{2}$ are the set of directed and



(a) ADMG $\mathcal{G}$. (b) $\mathcal{G}_1 \subseteq \mathcal{G}$. (c) $\mathcal{G}_2 \subseteq \mathcal{G}$.

Figure 1: (a) An example of a probabilistic ADMG $\mathcal{G}$ with corresponding edge probabilities. (b) and (c) are two different subgraphs of $\mathcal{G}$ in which $Q[y]$ is identifiable.

bidirected edges among the vertices, respectively. With slight abuse of notation, if $e \in E_d^{\mathcal{G}} \cup E_b^{\mathcal{G}}$, we write $e \in \mathcal{G}$. We use $\mathcal{G}' \subseteq \mathcal{G}$ when $\mathcal{G}'$ is an edge-induced subgraph of $\mathcal{G}$, i.e., $\mathcal{G}' = (V^{\mathcal{G}'}, E_d^{\mathcal{G}'}, E_b^{\mathcal{G}'})$, where $V^{\mathcal{G}'} = V^{\mathcal{G}}$ and $E_i^{\mathcal{G}'} \subseteq E_i^{\mathcal{G}}$ for $i \in \{b, d\}$. We denote by $\mathcal{G}[X]$ the vertex-induced subgraph of $\mathcal{G}$ over the subset of vertices $X \subseteq V^{\mathcal{G}}$. For a set of vertices X, we denote by $Anc_{\mathcal{G}}(X)$ the set of vertices in $\mathcal{G}$ that have a directed path to X. Note that $X \subseteq Anc_{\mathcal{G}}(X)$.

Definition 1 (Identifiability [13]). *Given a causal ADMG $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$, and two disjoint subsets of variables $X, Y \subseteq V^{\mathcal{G}}$, the causal effect of X on Y, denoted by $P_X(Y)$, is identifiable in $\mathcal{G}$ if $P_X^{M_1}(Y) = P_X^{M_2}(Y)$ for any two models M_1 and M_2 that induce $\mathcal{G}$ and $P^{M_1}(V^{\mathcal{G}}) = P^{M_2}(V^{\mathcal{G}}) > 0$.*

Definition 2 (Valid identification formula). *For a causal ADMG $\mathcal{G}$ over variables $V^{\mathcal{G}}$ and a causal query $P_X(Y)$, we say a functional $\mathcal{F}$ defined on the probability space over $V^{\mathcal{G}}$ is a valid identification formula for $P_X(Y)$ in $\mathcal{G}$ if $P_X^{M_1}(Y) = P_X^{M_2}(Y) = \mathcal{F}(P^{M_1}(V^{\mathcal{G}})) = \mathcal{F}(P^{M_2}(V^{\mathcal{G}}))$ for any two models M_1 and M_2 that induce $\mathcal{G}$ and $P^{M_1}(V^{\mathcal{G}}) = P^{M_2}(V^{\mathcal{G}}) > 0$.*

For post-interventional distribution $P_X(Y)$, let $[\mathcal{G}]_{Id(P_X(Y))}$ denote the set of ADMGs in which $P_X(Y)$ is identifiable. We denote by $Q[Y]$ the causal effect of $V \backslash Y$ on Y, i.e., $Q[Y] = P(Y|do(V \backslash Y))$.

Definition 3 (District [4]). *For ADMG $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$, let $\mathcal{G}_{\leftrightarrow}$ denote the edge-induced subgraph of $\mathcal{G}$ over its bidirected edges. $X \subseteq V^{\mathcal{G}}$ is a district (aka c-component) in $\mathcal{G}$ if $\mathcal{G}_{\leftrightarrow}[X]$ is connected.*

Definition 4 (Hedge [16]). *Let $\mathcal{G}$ be an ADMG, and $Y \subsetneq X$ be two subsets of its vertices, where Y is a district in $\mathcal{G}[Y]$. Vertices X form a hedge for $Q[Y]$ if X is a district in $\mathcal{G}[X]$ and $Anc_{\mathcal{G}[X]}(Y) = X$.*

Definition 5 (Maximal hedge [1]). *For ADMG $\mathcal{G}$ and a set of its vertices Y, let X be the union of all hedges formed for $Q[Y]$. Graph $\mathcal{G}[X]$, denoted by $\boldsymbol{MH}(\mathcal{G}, Y)$, is called the maximal hedge for $Q[Y]$.*

As an example, both sets $\{x, t\}$ and $\{z, t\}$ form a hedge for $Q[x]$ in $\mathcal{G}$ in Figure 1a, and $\mathcal{G}[\{x, z, t\}]$ is the maximal hedge for $Q[x]$.

2.1 Problem setup

Let $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$ be an ADMG, where $V^{\mathcal{G}}$ is the set of vertices each representing an observed variable of the system, $E_d^{\mathcal{G}}$ is the set of directed edges, and $E_b^{\mathcal{G}}$ is the set of bidirected edges among $V^{\mathcal{G}}$. We know *a priori* that the true ADMG describing the system is an edge-induced subgraph of $\mathcal{G}$,[1] and we are given a probability map that indicates for each subgraph of $\mathcal{G}$ such as $\mathcal{G}_s$, with what probability $\mathcal{G}_s$ is the true causal ADMG of the system. We denote this probability as $P(\mathcal{G}_s)$. For instance, if edge probabilities p_e are assumed to be mutually independent, $P(\mathcal{G}_s)$ takes the form:

$$P(\mathcal{G}_s) = \prod_{e \in \mathcal{G}_s} p_e \prod_{e \notin \mathcal{G}_s} (1 - p_e). \tag{1}$$

In what follows, we will refer to $P(\mathcal{G}_s)$ simply as the probability of the ADMG $\mathcal{G}_s$. The first problem of our interest is formally defined as follows.

Problem 1. *We consider the problem of finding the most probable edge-induced subgraph of $\mathcal{G}$, in which the causal effect $Q[Y]$ is identifiable. That is, the goal is to find the ADMG $\mathcal{G}^*$ defined by*

$$\mathcal{G}^* := \underset{\substack{\mathcal{G}_s \subseteq \mathcal{G}, \\ \mathcal{G}_s \in [\mathcal{G}]_{Id(Q[Y])}}}{\arg\max} \quad P(\mathcal{G}_s). \tag{2}$$

[1] Note that $\mathcal{G}$ can be a complete graph over both its directed and bidirected edges.

We will prove in Proposition 1 that if $Q[Y]$ is identifiable in $\mathcal{G}$, then it is also identifiable in every edge-induced subgraph of $\mathcal{G}$. In other words, if $\mathcal{G}$ is a feasible solution to the above optimization problem, so are all its edge-induced subgraphs. Furthermore, the same identification functional that is valid w.r.t. $\mathcal{G}$, is also valid w.r.t. every subgraph of $\mathcal{G}$. Let us illustrate this first on an example.

Example 1. *Consider the ADMG in Figure 1a. With the given edge probabilities and assuming independence among the edge probabilities, the subgraph of $\mathcal{G}$ illustrated in Figure 1b has probability $0.7 \times 0.7 \times 0.1 = 0.049$, whereas the subgraph of Figure 1c has probability $0.3 \times 0.3 \times 0.9 = 0.081$ (see Eq. (1)). If we were to solve Problem 1, we would choose $\mathcal{G}_2$ over $\mathcal{G}_1$, as it has a higher probability. Now consider identification formulas in $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively:*

$$\mathcal{F}_1: \quad Q[Y] = P(Y|X), \quad \mathcal{F}_2: \quad Q[Y] = \sum_{Z,T} P(Y|X,Z,T)P(Z,T).$$

$\mathcal{F}_1$ is a valid identification formula for any edge-induced subgraph of $\mathcal{G}_1$ (see Proposition 1). Analogously, $\mathcal{F}_2$ is valid for all edge-induced subgraphs of $\mathcal{G}_2$. If we consider the aggregate probability of the subgraphs of $\mathcal{G}_1$ and $\mathcal{G}_2$, i.e.,

$$\sum_{\hat{\mathcal{G}} \subseteq \mathcal{G}_1} P(\hat{\mathcal{G}}) = 1 - 0.9 = 0.1, \quad \textit{versus} \quad \sum_{\hat{\mathcal{G}} \subseteq \mathcal{G}_2} P(\hat{\mathcal{G}}) = (1-0.7) \times (1-0.7) = 0.09,$$

then we should prefer choosing $\mathcal{G}_1$ over $\mathcal{G}_2$, as its identification formula $\mathcal{F}_1$ is more likely to be valid than $\mathcal{F}_2$ considering the fact that for all its subgraphs, the identification functional $\mathcal{F}_1$ is still valid.

Plausibility of a certain identification functional $\mathcal{F}$ is the sum of the probability of all graphs in which $\mathcal{F}$ is valid given the query of interest. As we shall show Proposition 1, when an identification functional is valid in a causal graph, it is also valid in all its edge-induced subgraphs. We therefore propose the following problem as a surrogate to the problem of recovering the most plausible identification formula.

Problem 2. *Consider the problem of finding the edge-induced subgraph $\mathcal{H}^*$ of $\mathcal{G}$ with maximum aggregate probability of its subgraphs, in which $Q[Y]$ is identifiable. Formally,*

$$\mathcal{H}^* := \underset{\mathcal{G}_s \subseteq \mathcal{G},\, \mathcal{G}_s \in [\mathcal{G}]_{Id(Q[Y])}}{\arg\max} \sum_{\hat{\mathcal{G}} \subseteq \mathcal{G}_s} P(\hat{\mathcal{G}}). \tag{3}$$

In other words, we are looking for a graph $\mathcal{H}^*$ with the maximum aggregate probability of its subgraphs, among the graphs in $[\mathcal{G}]_{Id(Q[Y])}$, i.e., the graphs in which $Q[Y]$ is identifiable. Note that the objective which is maximised in Eq. 3 is a lower bound on the plausibility of an identification formula which is valid in $\mathcal{G}_s$. Therefore, Problem 2 is a surrogate to recovering the identification formula with the highest plausibility.

Remark 1. *It is noteworthy that our results are not limited to causal queries of the form $Q[Y] = P(Y|do(V^{\mathcal{G}} \setminus Y))$. They can be applied to general causal queries of the form $P_X(Y)$ if the directed edges of $\mathcal{G}$ and therefore the set $Anc_{\mathcal{G}\setminus X}(Y)$ are known. This is because the causal query $P_X(Y)$ can be expressed as $\sum_{Anc_{\mathcal{G}\setminus X}(Y)\setminus Y} Q[Anc_{\mathcal{G}\setminus X}(Y)]$, where $Anc_{\mathcal{G}\setminus X}(Y)$ is the set of ancestors of Y in $\mathcal{G}$ after removing the vertices of X. Furthermore, $P_X(Y)$ is identifiable in $\mathcal{G}$ if and only if $Q[Anc_{\mathcal{G}\setminus X}(Y)]$ is identifiable in $\mathcal{G}$ [19, 16, 9].*

In the sequel, for simplicity, we study Problems 1 and 2 under the following assumption. However, as proved in Appendix C, our results are valid in a more general setting where we allow only for perfect negative or positive correlations among the edges. An example of perfect negative correlation between two edges is that both of them cannot exist simultaneously.

Assumption 1. *The edges in $\mathcal{G}$ are mutually independent. That is, the probability of a subgraph $\mathcal{G}_s$ of $\mathcal{G}$ is of the form in (1).*

3 Reduction to Edge ID problem and Establishing Complexity

We begin this section with the following proposition, to which we referred before. Thereafter, we discuss the hardness of the two problems considered in this work.

Proposition 1. *For any causal query $P_X(Y)$ and ADMG $\mathcal{G}$, if $\mathcal{F}$ is a valid identification formula for $P_X(Y)$ in $\mathcal{G}$ (Def. 2), then $\mathcal{F}$ is a valid identification formula for $P_X(Y)$ in any $\mathcal{G}' \subseteq \mathcal{G}$.*

All proofs are presented in Appendix A. In what follows, we first formally define the edge ID problem, and then show the equivalence of Problems 1 and 2 to the edge ID problem under Assumption 1.

Definition 6 (Edge ID problem)**.** *For ADMG $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$, a set of non-negative edge weights $W_{\mathcal{G}} = \{w_e \geq 0 | e \in \mathcal{G}\}$, and a causal query $Q[Y]$ for a subset of variables $Y \subseteq V^{\mathcal{G}}$, the objective of the edge ID problem is to find the set of edges $E^* \subseteq E_d^{\mathcal{G}} \cup E_b^{\mathcal{G}}$ with minimum aggregated weight (cost), such that $Q[Y]$ is identifiable in the graph resulting from removing E^* from $\mathcal{G}$. Formally,*

$$\begin{aligned} E^* := & \underset{E \subseteq E_d^{\mathcal{G}} \cup E_b^{\mathcal{G}}}{\arg\min} \sum_{e \in E} w_e, \\ \text{s.t.} \quad & \mathcal{G}' = (V^{\mathcal{G}}, E_d^{\mathcal{G}} \setminus E, E_b^{\mathcal{G}} \setminus E) \in [\mathcal{G}]_{Id(Q[Y])}. \end{aligned} \tag{4}$$

We implicitly assume that the cost of removing a set of edges from $\mathcal{G}$ is the sum of the weights of each individual edge.

The following result unifies the two problems considered in this work by establishing their equivalence to the edge ID problem.

Lemma 1. *Under Assumption 1, Problem 1 is equivalent to the edge ID problem with the edge weights chosen to be the log propensity ratios, i.e., $w_e = \max\{0, \log(\frac{p_e}{1-p_e})\}$, $\forall e \in \mathcal{G}$. Moreover, Problem 2 is equivalent to the edge ID problem with the choice of weights $w_e = -\log(1 - p_e)$, $\forall e \in \mathcal{G}$. That is, an instance of Problems 1 and 2 can be reduced to an instance of the edge ID problem in polynomial time, and vice versa.*

As we mentioned earlier, the equivalence of these three problems can be established in more general settings than what is described under Assumption 1. We refer the interested reader to Appendix C for a discussion on one such setting. The following result shows that no polynomial-time algorithm for solving any of these three problems exists unless P = NP.

Theorem 1. *The edge ID problem is NP-hard.*

Theorem 1 is established through a reduction from the minimum vertex cover problem, which is known to be NP-hard [11]. Theorem 1 is a key result which shows the hardness of recovering the most plausible graph in which a specified causal effect of interest is identifiable.

Corollary 1. *Problems 1 and 2 are NP-hard under Assumption 1.*

It is noteworthy that the size of the problem depends on the number of vertices of $\mathcal{G}$, i.e., $|V^{\mathcal{G}}|$, and the number of edges of $\mathcal{G}$ with finite weight, i.e., $|E^{\mathcal{G}}| = |E_d^{\mathcal{G}}| + |E_b^{\mathcal{G}}|$. Since the ID algorithm [16] runs in time $\mathcal{O}(|V^{\mathcal{G}}|^2)$, the brute-force algorithm that tests the identifiability of $Q[Y]$ in every edge-induced subgraph of $\mathcal{G}$ and chooses the one with the minimum weight of deleted edges runs in time $\mathcal{O}(2^{|E^{\mathcal{G}}|}|V^{\mathcal{G}}|^2)$. In the next Section, we present various algorithmic approaches for solving or approximating the solutions to these problems.

4 Algorithmic Approaches

We first present a recursive approach for solving the edge ID problem in Section 4.1, described in Algorithm 1. Since the problem itself is NP-hard, Algorithm 1 runs in exponential time in the worst case. In Section 4.2, we present heuristic approximations of the edge ID problem which run in cubic time in the worst case. These heuristics can also be used as a pre-process to reduce the runtime of Alg. 1 by providing an upper bound which can be fed into Alg. 1 to prune the search space. Finally, in Section 4.3, we present a reduction of edge ID to yet another NP-hard problem, namely minimum-cost intervention problem [1], which allows us to use the algorithms designed for that problem to solve edge ID. Our simulations in Section 5 evaluate these approaches against each other.

4.1 Recursive exact algorithm

This approach is described in Algorithm 1. The inputs to the algorithm are an ADMG $\mathcal{G}$ along with edge weights $W_{\mathcal{G}}$, a set of vertices Y corresponding to the causal query $Q[Y]$, an upper bound ω^{ub} on the aggregate weight (cost) of the optimal solution, and a threshold ω^{th}, an upper bound on the acceptable cost of a solution. The closer ω^{ub} is to the optimal cost, the quicker Algorithm 1 will find

Algorithm 1 Recursive Algorithm for edge ID.

1: **function EDGEID**$(\mathcal{G}, Y, W_{\mathcal{G}}, \omega^{ub}, \omega^{th})$
2: $\mathcal{H} \leftarrow$ **MH**$(\mathcal{G}, Y)$
3: **if** $\mathcal{H} = \mathcal{G}[Y]$ **then return** $(True, \emptyset)$
4: $ID \leftarrow False$, $E_{min} \leftarrow \emptyset$
5: **while** True **do**
6: $e \leftarrow$ The edge of $\mathcal{H}$ with minimum weight
7: **if** $w_e = \infty$ **or** $w_e > \omega^{ub}$ **then return** (ID, E_{min})
8: $(id, E) \leftarrow$ **EDGEID**$(\mathcal{H} \setminus e, Y, W_{\mathcal{G}} \setminus \{w_e\}, \omega^{ub} - w_e, \omega^{th} - w_e)$
9: **if** $id = True$ **then**
10: $ID \leftarrow True$, $\omega_E \leftarrow w_e + \sum_{e_j \in E} w_{e_j}$
11: $\omega^{ub} \leftarrow \omega_E$, $E_{min} \leftarrow E \cup \{e\}$
12: **if** $\omega^{ub} \leq \omega^{th}$ **then return** (ID, E_{min})
13: Update $w_e \leftarrow \infty$ in $W_{\mathcal{G}}$

the solution. If no upper bound is known, the algorithm can be initiated with $\omega^{ub} = \infty$. However, we shall discuss a few approaches to find a good upper bound ω^{ub} in the following Section. Note that when $\omega^{th} = 0$, Algorithm 1 will output the optimal solution. Otherwise, as soon as a feasible solution with weight less than ω^{th} is found, the algorithm terminates (line 12).

The algorithm begins with calling subroutine **MH** in line 2, which constructs the maximal hedge for $Q[Y]$, denoted by $\mathcal{H}$. We discuss this subroutine in detail in Appendix B. Throughout the rest of the algorithm, we only consider the edges in $\mathcal{H}$, as the other edges do not alter the identifiability. If there is no hedge formed for $Q[Y]$, i.e., $\mathcal{H} = \mathcal{G}[Y]$, there is no need to remove any edges from $\mathcal{G}$ and the effect is already identified. Otherwise, we remove the edge with the lowest weight (e) from $\mathcal{H}$ and recursively call the algorithm to find the solution after removing the edge e, unless the weight of e is already higher than the upper bound ω^{ub}, which means no feasible solutions exist for the provided upper bound (line 7). Whenever a feasible solution is found, the upper bound ω^{ub} is updated to the lowest weight among all the solutions weights discovered so far (line 11). This in turn helps the algorithm prune the exponential search space during the next iterations to reduce the runtime. As soon as a solution with a weight less than the acceptable threshold, i.e., ω^{th}, is found, the algorithm returns the solution. Otherwise, w_e is updated to infinity so that it never gets removed (line 13). This is due to the fact that we have already explored all the solutions involving e.

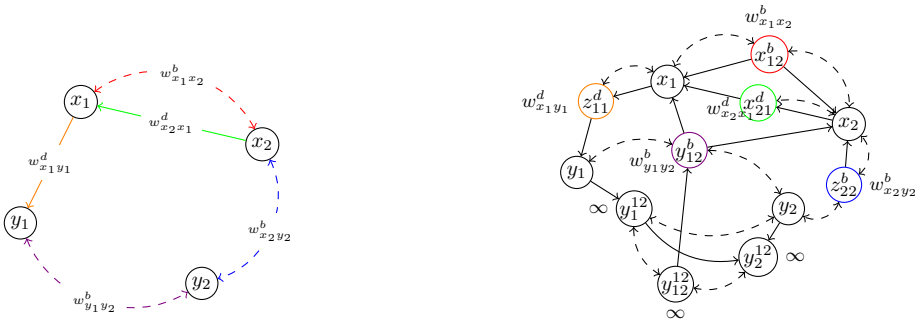
4.2 Heuristic algorithms

In this Section, we present two heuristic algorithms for approximating the solution to the edge ID problem. These algorithms can also be used to find the upper bound ω^{ub} efficiently, which could be fed as an input to Algorithm 1.

Algorithm 2 Heuristic algorithm for Edge ID.

function HEID$(\mathcal{G}, Y, W_{\mathcal{G}})$
$\mathcal{G}' \leftarrow$ **MH**$(\mathcal{G}, Y)$
$Z \leftarrow \{z \in V^{\mathcal{G}'} | \exists y \in Y : \{z, y\} \in E_b^{\mathcal{G}'}\} \setminus Y$
$\mathcal{H} \leftarrow$ The induced subgraph of $\mathcal{G}'$ on its directed edges.
$W_{\mathcal{H}} \leftarrow \{w_e \in W_{\mathcal{G}} | e \in \mathcal{H}\}$, $V^{\mathcal{H}} \leftarrow V^{\mathcal{H}} \cup \{y^*, z^*\}$
for $z \in Z$ **do** $E^{\mathcal{H}} \leftarrow E^{\mathcal{H}} \cup (z^*, z)$, $W_{\mathcal{H}} \leftarrow W_{\mathcal{H}} \cup \{w_{(z^*,z)} = \sum_y w_{\{z,y\}}\}$
for $y \in Y$ **do** $E^{\mathcal{H}} \leftarrow E^{\mathcal{H}} \cup (y, y^*)$, $W_{\mathcal{H}} \leftarrow W_{\mathcal{H}} \cup \{w_{(y,y^*)} = \infty\}$
$E \leftarrow MinCut(\mathcal{H}, W_{\mathcal{H}}, z^*, y^*)$
return $(E, \sum_{e \in E} w_e)$

Let $Z = \{z \in V^{\mathcal{G}} | \exists y \in Y : \{z, y\} \in E_b^{\mathcal{G}}\} \setminus Y$ denote the set of vertices that have at least one common bidirected edge with a vertex in Y. Any hedge formed for $Q[Y]$ contains at least one vertex of Z. As a result, in order to eliminate all the hedges formed for $Q[Y]$, it suffices to make sure that none of the vertices in Z appear in such a hedge. To this end, for any $z \in Z$, it suffices to either



(a) ADMG $\mathcal{G}$, $Y = \{y_1, y_2\}$.

(b) ADMG $\mathcal{H}$, $Y^{mcip} = \{y_1, y_2, y_2^{12}\}$.

Figure 2: Reduction from edge ID to MCIP.

remove all the bidirected edges between z and Y, or eliminate all the directed paths from z to Y. The problem of eliminating all directed paths from Z to Y can be cast as a minimum cut problem between Z and Y in the edge-induced subgraph of $\mathcal{G}$ over its directed edges. To add the possibility of removing the bidirected edges between Z and Y, we add an auxiliary vertex z^* to the graph, and draw a directed edge from z^* to every $z \in Z$ with weight $w = \sum_{y \in Y} w_{\{z,y\}}$, i.e., the sum of the weights of all bidirected edges between z and Y. Note that z can have bidirected edges to multiple vertices in Y. We then solve the minimum cut problem for z^* and Y. If an edge between z^* and $z \in Z$ is included in the solution to this minimum cut problem, it is mapped to removing all the bidirected edges between z and Y in the main problem. Note that we can run the algorithm on the maximal hedge formed for $Q[Y]$ in $\mathcal{G}$ rather than $\mathcal{G}$ itself. This heuristic is presented as Algorithm 2.

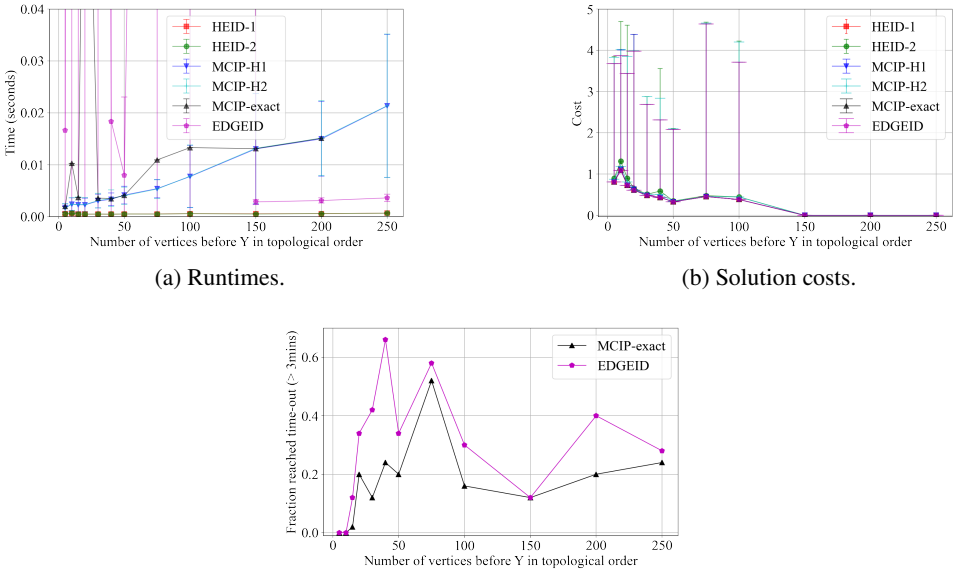
An analogous approach which goes through solving an undirected minimum cut on the edge induced subgraph of $\mathcal{G}$ over its bidirected edges is presented in Algorithm 4 in Appendix D. As mentioned earlier, these algorithms can be used either as standalone algorithms to approximate the solution to the edge ID problem, or as a pre-processing step to find an upper bound ω^{ub} for Algorithm 1. As we shall see in our simulations, both algorithms achieve near-optimal results on random graphs.

4.3 Alternative approach: reduction to MCIP

As an alternative approach to the algorithms discussed so far, we present a reduction of the edge ID problem to another NP-hard problem, i.e., the minimum-cost intervention problem (*MCIP*) introduced in [1]. This reduction allows us to exploit algorithms designed for MCIP to solve our problems. The formal definition of MCIP is as follows.

Definition 7 (MCIP). *Suppose $\mathcal{G} = (V^{\mathcal{G}}, E_d^{\mathcal{G}}, E_b^{\mathcal{G}})$ is an ADMG, $C : V^{\mathcal{G}} \to \mathbb{R}^{\geq 0}$ is a cost function mapping each vertex of $\mathcal{G}$ to a non-negative cost, and $Y \subseteq V^{\mathcal{G}}$. The objective of the minimum-cost intervention problem for identifying the causal effect $Q[Y]$ is to find the subset $A \subseteq V^{\mathcal{G}}$ with the minimum aggregate cost such that $Q[Y]$ is identifiable in $\mathcal{G} \setminus A$, that is, the graph that results from $\mathcal{G}$ after removing the vertices in A and the edges attached to them.*

The reduction from edge ID to MCIP is based on a transformation from ADMG $\mathcal{G}$ to another ADMG $\mathcal{H}$, where each edge in $\mathcal{G}$ is represented by a vertex in $\mathcal{H}$. This transformation is based on the causal query $Q[Y]$, and it maps the identifiability of $Q[Y]$ in $\mathcal{G}$ to identifiability of $Q[Y^{mcip}]$ in $\mathcal{H}$, where Y^{mcip} is a set of vertices in $\mathcal{H}$. This transformation satisfies the following property; removing a set of edges E^* in $\mathcal{G}$ makes $Q[Y]$ identifiable if and only if intervening on the corresponding vertices of E^* in $\mathcal{H}$ makes $Q[Y^{mcip}]$ identifiable. More precisely, after this transformation, solving the edge ID problem for $Q[Y]$ in $\mathcal{G}$ is equivalent to solving MCIP for $Q[Y^{mcip}]$ in $\mathcal{H}$. The complete details of this transformation can be found in Appendix A.2. An example of this reduction is shown in Figure 2, where $Q[\{y_1, y_2\}]$ in $\mathcal{G}$ (Figure 2a) is mapped to $Q[\{y_1, y_2, y_2^{12}\}]$ in $\mathcal{H}$ (Figure 2b). The vertices of $\mathcal{H}$ corresponding to each edge in $\mathcal{G}$ are indicated with the same color and have the same weight (cost). To avoid intervening on the remaining vertices in $\mathcal{H}$, we assign infinity cost to them. It is straightforward to see that the solution to the edge ID problem in $\mathcal{G}$ with the query $Q[Y = \{y_1, y_2\}]$ would be to remove the edge with the lowest weight. This is because after removing any edge in $\mathcal{G}$, no hedge



(a) Runtimes.

(b) Solution costs.

(c) Fraction for which runtime of 3 minutes exceeded.

Figure 3: Experimental results for runtime, solution costs, fraction of graphs for which no solution was found, and fraction of graphs for which runtime limit of 3 minutes was exceeded. Error bars for runtime and cost graphs indicate 5th and 95th percentiles. Best viewed in color.

remains for $Q[Y]$. Similarly, in $\mathcal{H}$, the solution to MCIP with the query $Q[Y^{mcip} = \{y_1, y_2, y_2^{12}\}]$ is to intervene on the vertex with the lowest cost among $Z = \{z_{11}^d, x_{21}^d, x_{12}^b, y_{12}^b, z_{22}^b\}$. This is because after intervening on any vertex in Z, no hedge remains for $Q[Y^{mcip}]$.

The following result formally establishes the link between the edge ID problem in $\mathcal{G}$ and MCIP in $\mathcal{H}$.

Proposition 2. *There exists a polynomial-time reduction from edge ID to MCIP and vice versa.*

5 Experiments

We evaluate the proposed heuristic algorithms 2 (HEID-1) and 4 (HEID-2), as well as the exact algorithm 1 (EDGEID), where the upper-bound ω^{ub} for EDGEID is set to be the minimum cost found by HEID-1 or -2. Furthermore, given the reduction of the edge ID problem to the MCIP problem described in Section 4.3, we also evaluate the two approximation and one exact algorithms from [1] (MCIP-H1, MCIP-H2, and MCIP-exact, respectively). All experiments were carried out on an Intel i9-9900K CPU running at 3.6GHz.

Simulations: The algorithms are evaluated for graphs with between 5 and 250 vertices. For a given number of vertices, we uniformly sample 50 ADMG structures from a library of graphs which are non-isomorphic to each other. Edges for each of these 100 graphs are sampled with probability of $\log(n)/n$ to impose sparsity (thus pragmatically reducing the search space). For each graph we sample directed and bidirected edge probabilities p_e uniformly between 0.51 and 1.0. The problem is then converted into edge ID according to Lemma 1. The vertices in the graphs are topologically sorted and the outcome Y is selected to be the last vertex in the topological ordering. We then check whether a solution exists in principle by removing all finite cost edges and checking for identifiability. If not, a new graph is sampled to avoid evaluating the algorithms on graphs with no solution. For each of these 50 probabilistic ADMGs, we run the algorithms and record the resulting runtime and the associated cost of the solution. If the runtime exceeds 3 minutes, we abort and log that the algorithm has failed to find a solution.

Results are presented in Figure 3. Runtimes and costs are shown for the subset of graphs for which all algorithms found a solution (to facilitate comparison). Runtimes for each algorithm are shown in Fig. 3a, where it can be seen that our proposed HEID-1 and HEID-2 heuristic algorithms have negligible runtime, followed by the MCIP variants. Interestingly, the exact algorithm EDGEID

outperformed the MCIP algorithms on larger graphs, for which the transformation time from the edge ID problem to the MCIP increases with the size of the graph. In contrast, EDGEID had large runtime variance which depended heavily on the specifics of the graph under evaluation, particularly for graphs with fewer vertices. The costs for each graph are shown in Fig. 3b, and here we see, as expected, the lowest cost is achieved by the two exact algorithms, EDGEID and MCIP-exact, followed closely by the heuristic algorithms. Figure 3c shows the fraction of evaluations for which the runtime exceeded 3 minutes (applicable to the exact algorithms). In general, and owing to the sparsity penalty in our graph generation mechanism, the cost of identified solutions falls with the number of vertices. However, among the exact algorithms, EDGEID, exceeds the 3 minute runtime more often than the MCIP-Exact, regardless of the number of vertices and despite the fact that EDGEID is quicker at finding a solution when it does so. Overall, HEID-1 was both the most consistent in terms of finding a solution, having a short runtime, and achieving a close to optimal cost.

Real-World Graphs: We also apply the algorithms to four real-world datasets. The first 'Psych' (22 nodes & 70 directed edges) concerns the putative structure from a causal discovery algorithm Structural Agnostic Model [10] using data collected as part of the Health and Relationships Project [20]. The other three 'Barley' (48 nodes & 84 directed edges), 'Water' (32 nodes & 66 directed edges), and 'Alarm' (37 nodes & 46 directed edges) come from the bnlearn python package [17]. For all four graphs, and as with the simulations described above, we introduce bidirected edges with a sparsity constraint of $\log(n)/n$, and simulate expert domain knowledge by random assigning directed and bidirected edge probabilities between 0.51 and 1. The outcome Y is selected to be the last vertex in the topological ordering. For these results, we provide the runtime (limited to 500 seconds) and cost, as well as the ratio of graph plausibility before and after selecting a subgraph in which the effect is identifiable $P(\hat{\mathcal{G}}^*)/P(\mathcal{G})$. This ratio is 1.0 if the effect is identifiable in the original graph, and decreases according to the plausibility of an identified subgraph.

Results are shown in Table 1. In cases where MCIP-exact and/or EDGEID did not exceed the runtime limit, it can be seen that HEID-2 and MCIP-H2 achieved equivalent to optimal cost and ratio. Runtimes for MCIP variants exceeded the HEID variants owing to the required transformation. EDGEID timed out on all but the Alarm structure, whereas MCIP-exact only timed out on the Psych structure, indicating that the MCIP-exact is more consistent (this also corroborates Figure 3c).

Table 1: Time (seconds), cost, and ratio $P(\hat{\mathcal{G}}^*)/P(\mathcal{G})$ for seven algorithms over four real-world datasets. A dash - indicates maximum runtime (500 seconds) exceeded.

Algorithm	**Psych**			**Barley**			**Alarm**			**Water**		
	Time	Cost	Ratio	Time	Cost	Ratio	Time	Cost	Ratio	Time	Cost	Ratio
HEID-1	0.0019	2.648	0.07	0.0026	0.081	0.92	0.0004	0.0	1.0	0.0019	1.02	0.36
HEID-2	0.0019	1.806	0.16	0.0026	0.081	0.92	0.0003	0.0	1.0	0.0017	0.42	0.66
MCIP-H1	0.0136	2.648	0.07	0.0140	0.081	0.92	0.0027	0.0	1.0	0.0124	1.02	0.36
MCIP-H2	0.0133	1.806	0.16	0.0131	0.081	0.92	0.0029	0.0	1.0	0.0113	0.42	0.66
MCIP-exact	-	-	-	0.0099	0.081	0.92	0.0028	0.0	1.0	0.0221	0.42	0.66
EDGEID	-	-	-	-	-	-	0.0005	0.0	1.0	-	-	-

6 Conclusion

In causal inference, researchers are often faced with graphs for which the effect of interest is not identifiable. It is common to identify a target effect by assuming ignorability (i.e., no unobserved confounders exist). A less drastic and more reasonable approach would be to relax this assumption by identifying the most plausible subgraph, given uncertainty about the structure as we have suggested in this work. We presented a number of algorithms for finding the most probable/plausible probabilistic ADMG in which the target causal effect is identifiable. We provided an analysis of the complexity of the problem, and provided an experimental comparison of runtimes, solution costs, and failure rates. We noted that one of our heuristic algorithms, HEID-1 (Alg. 2) and (Alg. 4) performed remarkably well across all metrics. In terms of limitations, we make the assumption that the edges in $\mathcal{G}$ are mutually independent (Assumption 1). Future work should explore the case where this assumption does not hold. Finally, it is worth remembering that the external validity of the derived subgraph (i.e., whether or not the subgraph is correctly specified with respect to the corresponding real-world process) is not guaranteed. As such, practitioners that use such approaches are encouraged to do so with caution, in particular for research involving human participants.

References

[1] S. Akbari, J. Etesami, and N. Kiyavash. Minimum cost intervention design for causal effect identification. *arXiv preprint*, arXiv:2205.02232, 2022.

[2] E. Bareinboim, J.D. Correa, D. Ibeling, and T. Icard. On Pearl's hierarchy and the foundations of causal inference. *ACM Special Reports*, 2020.

[3] A. Deaton and N. Cartwright. Understanding and misundertstanding randomized controlled trials. *Social Science and Medicine*, 210:2–21, 2018. doi: 10.1016/j.socscimed.2017.12.005.

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

[5] C. Glymour, K. Zhang, and P. Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019.

[6] Y. Huang and M. Valtorta. Pearl's calculus of intervention is complete. *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence (UAI)*, 2006. doi: 10.5555/3020419.3020446.

[7] G.W. Imbens and J.D. Angrist. Identification and estimation of local average treatment effects. *Econometrica*, 62(2):467–475, 1994. doi: 10.2307/2951620.

[8] G.W. Imbens and D.B. Rubin. *Causal inference for statistics, social, and biomedical sciences. An Introduction.* Cambridge University Press, New York, 2015.

[9] Amin Jaber, Jiji Zhang, and Elias Bareinboim. Causal identification under markov equivalence: Completeness results. In *International Conference on Machine Learning*, pages 2981–2989. PMLR, 2019.

[10] D. Kalainathan, O. Goudet, I. Guyon, D. Lopez-Paz, and M. Sebag. Structural agnostic modeling: Adversarial learning of causal graphs. *arXiv:1803.04929v3*, 2020.

[11] Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

[12] J. Pearl. Aspects of graphical models connceted with causality. *Proceedings of the 49th Session of the International Statistical Institute*, pages 399–401, 1993.

[13] J. Pearl. *Causality*. Cambridge University Press, Cambridge, 2009.

[14] D. B. Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005. doi: 10.1198/016214504000001880.

[15] R. D. Shah and J. Peters. The hardness of conditional independence testing and the generalised covariance measure. *The Annals of Statistics*, 48(3), 2020.

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

[17] E. Taskesen. bnlearn - Library for Bayesian network learning and inference. *Python Library*, 2020. URL `https://erdogant.github.io/bnlearn`.

[18] J. Tian and J. Pearl. A general identification condition for causal effects. *AAAI*, 2002.

[19] Jin Tian and Judea Pearl. On the testable implications of causal models with hidden variables. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 519–527, 2002.

[20] D. Umberson. Health and relationships project. *Inter-university consortium for political and social research*, 2014-2015. doi: 10.3886/ICPSR37404.v2.

[21] M. J. van der Laan and S. Rose. *Targeted Learning - Causal Inference for Observational and Experimental Data*. Springer International, New York, 2011.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] ...
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]